A MULTI-LAYER SWARM CONTROL MODEL FOR INFORMATION PROPAGATION AND MULTI-TASKING

A Dissertation
Presented to
The Academic Faculty

By

Said Salim Hamdan Al-Abri

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology

August 2019

A MULTI-LAYER SWARM CONTROL MODEL FOR INFORMATION PROPAGATION AND MULTI-TASKING

Approved by:

Dr. Fumin Zhang, Advisor
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Magnus Egerstedt
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Yorai Wardi
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Molei Tao
School of Mathematics
Georgia Institute of Technology

Dr. Samuel Coogan
School of Electrical and Computer Engineering
Georgia Institute of Technology

Date Approved: June 17, 2019

Discovery is seeing what everybody else has seen, and thinking what nobody else has thought.

Albert Szent-Györgyi

To my great wife Zainab

and my brilliant children Ibrahim and Lateen

without whom this work would have been impossible.

ACKNOWLEDGEMENTS

I would like to take this opportunity to express my greatest appreciation to people who have helped and supported me during my life as a Ph.D. student.

The very first people I would like to thank are my family. Special gratitude goes out to my wife Zainab, and my kids Ibrahim and Lateen. I cannot reward them for all the time that I stole from their life during my Ph.D. journey, neither I can reward them for all the overwhelming love, support, and patience that they granted me. I am also grateful to my parents in Oman who withstand being away from them for such a long time, and who have been my model for endless hardworking.

I also give my deep thanks to Prof. Fumin Zhang, my advisor, for his support, guidance, and encouragement during my Ph.D. journey. In particular, he taught me how to aim higher and what really matters to be a successful researcher.

I would like to thank my fellow doctoral students for their feedback, cooperation, and help. In particular, I want to thank my colleague Sean Maxon for all the brainstorming and discussions we have together. I also want to thank my colleagues Tony Lin, Mengxue Hou and Qiuyang Tao who helped me running blimp experiments at the Georgia Tech Systems Research Lab, and Sean Wilson who helped me running experiments at Georgia Tech Robotarium.

Finally, I would like to thank my thesis committee members: Prof. Yorai Wardi, Prof. Magnus Egerstedt, Prof. Molei Tao and Prof. Samuel Coogan, for their precious time and valuable suggestions for the work done in this dissertation.

TABLE OF CONTENTS

LIST OF FIGURES

SUMMARY

Modeling and control of multi-agent systems is an important problem due to its large variety of potential applications and increasing practical and theoretical challenges. A large part of inspiration for modeling and control of multi-agent systems originates from the study of natural collective behaviors observed for example in schools of fish, flocks of birds, colonies of ants and cultures of bacteria. While individuals in these natural swarms are collectively performing complex tasks such as foraging or synchronization, critical information such as predator warnings propagate across the swarm almost instantly and presumably without explicit communication between the individuals.

On the other hand, algorithms for multi-agent systems to locate a source or to follow a desired level curve of spatially distributed scalar fields generally require sharing field measurements among the agents for gradient estimation. The dependence on the exchange of data through a communication channel is a hard requirement that might be undesired especially in applications with severe limitations such as underwater robotics.

The main contribution of this Dissertation is a Multi-Layer control model composed of an interplay of decentralized algorithms for perception and swarming. In the perception layer, each agent applies a Principal Component Analysis (PCA) on the relative positions and headings of its neighbors to learn principal properties about the motion and the geometry of the spatial distribution of the surrounding agents. These principal components are then used in the swarming layer where various distributed control laws are designed to balance between achieving a collective task and at the same time allowing critical emerging signals to propagate to the entire swarm.

Within this Multi-Layer model, we contributed distributed control laws for swarms to perform collective source seeking and level curve tracking of scalar fields. These control laws scale to swarms of various sizes and graph structures and do not rely on explicitly estimating the field gradient or explicitly sharing measurements among the agents.

Additionally, we contributed a distributed control law that balances between achieving a collective task and at the same time allowing critical signals to propagate to the entire swarm. Through this, we demonstrated implicit information propagation in swarms exhibiting predator-avoidance behavior using only local interactions and without explicit communication or prescribed formations. Moreover, we obtained various stability results reflecting the convergence and robustness of the proposed algorithms. Finally, we validated the proposed model for source seeking, level curve tracking and predator avoidance behaviors through various simulation and experimental results.

The proposed control model offers a new method that enables robots with limited resources to perform diverse swarming activities with only local information. Additionally, designing analytical models to understand information propagation will not only reveal natural mysteries but additionally will help to propose multi-tasking control algorithms for robotic swarms that require only very limited or no explicit communication.

CHAPTER 1
INTRODUCTION

Modeling and control of multi-agent systems is an important problem due to its large variety of potential applications and increasing practical and theoretical challenges. A large part of inspiration for modeling and control of multi-agent systems originates from the study of natural collective behaviors observed for example in schools of fish, flocks of birds, colonies of ants and cultures of bacteria. A remarkable feature in these natural collective behaviors is their emergence from presumably pure local interactions [1, 2, 3]. This potentially guides us in using swarms of robots in solving complex problems with limited sensing, processing and communicating capabilities [4].

Swarm robotics is a multi-agent system composed of a large number of robots which are often simple and small. Agents in the swarm interact with each other and with the environment leading to emergent collective behavior. A behavior is considered emergent when it appears indirectly and unexpectedly from simple local interactions. Agents in a swarm coordinate their motion based on the locally available information. The information is considered to be locally accessible when agents obtain them via measurements of their own sensors or via communication channels with the neighboring agents. Examples of sensor-based local information include inter-agent distances, relative positions, and bearing angles [5].

In this Dissertation, we propose a Multi-Layer control model composed of an interplay of decentralized algorithms for perception and swarming as shown in Fig. 1.1. In the perception layer, each agent applies a Principal Component Analysis (PCA) on the relative positions and headings of its neighbors to learn principal properties about the motion and the geometry of the spatial distribution of the surrounding agents. These principal components are then used in the swarming layer where various distributed control laws are

designed to balance between achieving a collective task and at the same time allowing critical emerging signals to propagate to the entire swarm. The collective behaviors we consider in this Dissertation are source seeking, level curve tracking, turning wave propagation, and predator avoidance.

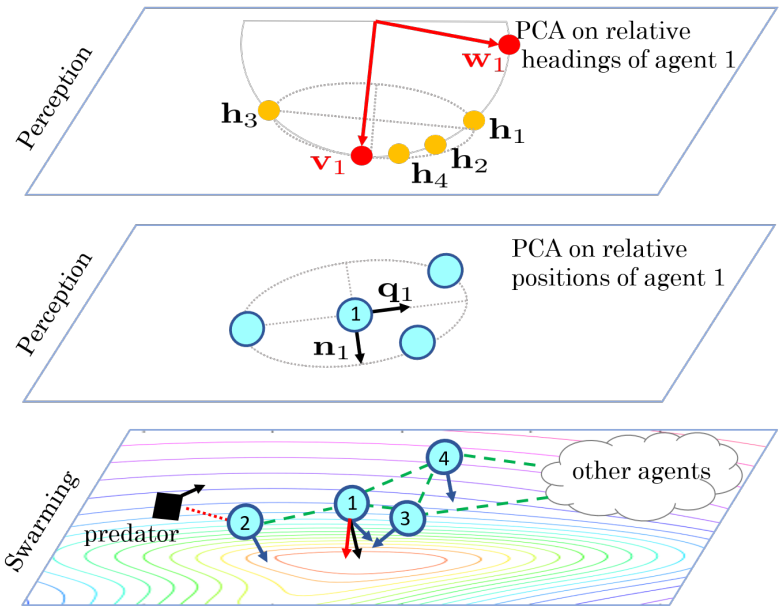


Figure 1.1: The proposed Multi-Layer control model. The principal geometrical components ($\mathbf{n}$,$\mathbf{q}$,$\mathbf{v}$,$\mathbf{w}$) obtained by the perception layers are used in the swarming layer to coordinate the motion of each agent.

Source seeking and level curve tracking are important swarm robotics problems where multiple robots are deployed to locate a source or track a desired level curve of a scalar field. The source is the location where the field has a maximum or minimum value. A level curve of a scalar field with a non-vanishing gradient is a curve consisting of all points where the field has the same value. The field can represent an environmental characteristic including but not limited to, a chemical concentration, a light intensity, or a temperature value. These two problems have various applications ranging from monitoring environmental characteristics to positioning of source signals, to exploring and establishing hazard boundaries, and to search and rescue tasks, just to name a few [6, 7].

The various multi-agent control laws developed in the literature to solve these two

problems generally incorporate a mixture of field gradient and Hessian estimation, extremum seeking control, sliding-mode control, and weighted consensus laws. Most of the aforementioned control strategies either rely on sharing measurements via communication channels, require maintaining specific spatial formations, or apply only to certain sizes and structures of the interacting graphs. The dependence on the exchange of data through a communication channel is a hard requirement that might be undesired especially in applications with severe limitations such as underwater robotics.

In this Dissertation, the first contribution is extending an existing 3-dimensional gradient-free source seeking strategy from a 3-agent system into M-agent system [8, 9]. For this, we developed a two-layer system composed of social and environmental layers. In the social layer, the agents interact with each other by means of implicit and explicit consensus and formation control laws. In the environmental layer, the agents interact with the environment by modulating their speed as a function of the field value. Although this model does not rely on explicit communications, it requires some agents to be more capable of exploring the environment than the others.

The second contribution is replacing the social layer by a perception layer and generalizing the environmental layer to a swarming layer [10, 11, 12]. In the perception layer, each agent learns from the spatial distribution of its neighbors principal directions for motion. These directions are then used in the swarming layer to modulate velocities based on the environmental field value. Remarkably, through the interplay of the learning and swarming algorithms, swarms of various sizes and graph structures are able to perform collective source seeking and level curve tracking of scalar fields without the need to explicitly estimate the field gradient or explicitly share measurements among the agents.

The novel concept for our solution to the source seeking and the level curve tracking problems is in integrating a PCA learning algorithm [13, 14, 15] in the perception layer of the Multi-Layer model through which each agent locally obtains a body frame. This time-varying body frame is then used in the swarming layer by each agent to modulate its

motion based only on its instantaneous measurement of the field.

On the other hand, biological research has established evidence that critical decisions such as performing a turn in a flock of birds or avoiding a predator in a school of fish are initiated by individuals at the boundary of the swarm and then propagate to the entire collective [2, 3, 1]. The propagating signal might take the form of a change of motion behavior of an agent to alert the swarm about a predator or guide the swarm to a possible source of food, or to a shelter such as a dark area, as in the case of shiner fish schools [16]. The question is how this information propagation occurs almost instantly and presumably without explicit communication between the individuals even when they are significantly far away from where the signal starts. Even more surprising is how these external signals, for example, predator warnings, occur and propagate while the individuals are collectively performing other complex tasks such as foraging or synchronization.

In this Dissertation, the third contribution is adding to the Multi-Layer model an extra layer for perception [17]. In the perception layers, each agent applies not only PCA on the relative positions of its neighbors, but additionally, each agent applies PCA on relative headings of its neighbors. These learning algorithms are used to learn principal properties about the geometry and motion of the surrounding agents. The resulting principal components are then used in the swarming layer where a distributed control law is designed to balance between achieving a collective task and at the same time allowing critical emerging signals to propagate to the entire swarm. Through this novel model, we demonstrate implicit information propagation and multi-tasking in swarms using only local interactions and without explicit communication.

Our proposed model for information propagation treats sudden changes in the motion of individuals as a piece of valuable information. These changes often are viewed by existing models as disturbances that are suppressed by firm consensus or formation control laws for the sake of rigid synchronization or stability. We design the motion controllers to autonomously balance between synchronizing with neighbors and responding to internal

or external stimuli. Most of the existing results use the Vicsek model [18] to model the propagation of turning waves. However, authors in [19, 20] argue that wave propagation using the Vicsec model get attenuated and might not reach the entire swarm. In this paper, we develop a synchronizing algorithm based on PCA of locally observed headings. This algorithm tends to pay more attention to the outlying heading, which is different than the common consensus-on-a-sphere [9, 21] that converges to the average heading. This results in an agile and flexible motion behavior where individuals promptly respond to stimuli while achieving a collective task.

The successful elimination of the challenging requirement of explicit estimation and communication is attributed to the locally computed PCA-based body frame and the design of the control law. In particular, the PCA learning algorithm captures changes on the spatial shape and orientation of the swarm which represents an indirect feedback signal of how the field is affecting the motions of other agents. Additionally, the PCA learning algorithm is local since agents only require knowledge of relative positions. Complicating the convergence analysis, the PCA learning algorithm runs on a time scale that is different and faster than that of the swarming algorithm. To overcome this difficulty of multi-scale evolving time, we exploit a singular perturbation framework where the dynamics of the PCA learning and swarming control are viewed as rapidly decaying and slowly varying dynamics, respectively.

In all of the aforementioned contributions, we obtain various stability results reflecting the convergence and robustness of the proposed algorithms. The main difficulty in obtaining these results lays in the fact that we are proving emergent behavior that we don't explicitly control. This requires complicated derivations and manipulations in order to obtain implicit dynamics that describe the desired emergent behaviors. Additionally, most of the existing source seeking and level curve tracking works ignore the higher order terms of the field when either estimating the field gradient or analyzing the convergence. In this Dissertation, when the graph is complete, we do consider the entire field components which

allow us to realize their effect on the convergence and robustness of the algorithms.

Moreover, we validated the proposed model for numerous source seeking, level curve tracking and predator avoidance behaviors through various simulation and experimental results. The experiments are conducted using the Georgia Teach Robotarium [22] and the Georgia Teach Miniature Blimps [23]. These results suggest the efficiency, scalability, and robustness of the proposed model under different environments and robotic platforms.

The proposed control model offers a new method that enables robots with limited resources to perform various swarming activities with only local information. Additionally, designing analytical models to understand information propagation will not only reveal natural mysteries but additionally will help to propose multi-tasking control algorithms for robotic swarms that require only very limited or no explicit communication. In particular, this is highly related to the problem of designing tactics for a swarm of drones to avoid or chase a malicious agent [24, 25].

The rest of the dissertation is organized as follows. We first present background and related work in Chapter 2. Next, in Chapter 3, we present a distributed gradient-free source seeking in a 3D environment using a consensus-on-a sphere control law. Then in Chapter 4, we present the Multi-Layer model to develop a distributed gradient-free and communication-free source seeking strategy in 2D and $3D$ environments. This model is extended in Chapter 5 to develop a distributed gradient-free and communication-free level curve tracking. In Chapter 6, we present the overall Multi-Layer used for information propagation and multi-tasking. Finally, in Chapter 7, concluding remarks and future research possibilities are provided.

CHAPTER 2
BACKGROUND

In this chapter, we present the basic background and recent works related to the multi-agent source seeking, level curve tracking and information propagation in natural swarms. We also discuss the related work to the main concepts used to solve the aforementioned problems. This includes consensus-on-a sphere, flow models for principal component analysis, and local information for multi-agent systems.

2.1 Multi-Agent Source Seeking

There exists a huge amount of work related to the source seeking problem. Here we limit the scope to the multi-agent solutions with more focus on the recent attempts to avoid sharing field measurements.

In [26], assuming knowledge of the gradient, a distributed strategy is designed to climb the gradient as well as to achieve some desired formations. In [27], a distributed control law is developed to form a circular formation. The agents share their field measurements via a communication channel and then estimate the gradient and perform gradient climbing. Without knowing the global position of agents, the authors in [28] developed a distributed source seeking algorithm that relies on communicating field measurements to estimate the gradient. A different gradient-based strategy is studied in [29] where there are multiple sources and the agents are controlled such that the swarm autonomously splits into subgroups that each steers toward a source. A bio-inspired source seeking strategy called the Speeding Up and Slowing Down (SUSD), is developed in [30] and [31] in 2 and 3 dimensions, respectively. Although the agents do not share field measurements, they still need to share coordination measurements for a swarm of more than 2 agents in a 2-dimensional space and 3 agents in a 3-dimensional space. Reliable communication channels cannot

be always guaranteed especially for robots with severe resource limitations such as underwater robotics. Consequently, solutions that are independent of estimating the gradient or communicating measurements are of extreme importance.

Extremum-based source seeking approaches are developed intensively in the literature. The extremum seeking control is a model-free real-time optimization which is used to implicitly estimate the gradient of the field based on the instantaneous measurements of the field [32]. In [33] and [34], extremum-based source seeking control techniques are developed for a single vehicle for 2-dimensional and 3-dimensional spaces, respectively. The concept is to set the forward velocity as a constant while tune angular velocities based on an extremum seeking control. Although the approach is simple to implement, the vehicle needs a relatively long distance to travel until the gradient estimation improves. In [35] and [36], they extended the extremum-based source seeking to the multi-agent case. However, in this case, agents need to share their estimates.

In [37] and [38], they developed a strategy for a large number of robots with a complete graph that is independent of both gradient estimation and measurements communication. The strategy is based on a weighted consensus and a Gaussian perturbation such that the consensus value drifts toward the direction of the gradient. The main drawback is that the swarm drifts extremely slow toward the source.

2.2 Multi-agent Level Curve Tracking

Similar to the source seeking problem, the level curve tracking problem generally depends on estimating the field gradient which requires sharing of field measurements. In [39] and [40], they designed mission control laws that achieve source seeking and level curve tracking. The mission control laws are based on estimating the gradient and communicating measurements as well as maintaining specific formations. Alternatively, in [41], a strategy is designed for a 2-agent system to track the desired level curve, while maintaining a specific distance. The strategy is independent of gradient estimation, however, it requires

communicating field measurements. A discontinuous sliding mode control law is designed for single and multi-agent systems in [42] and [43] which is independent of estimating the field gradient and communicating field measurements. However, the multiple agents are only used for a formation control that forces the robots to spread along the level curve.

2.3 Information Propagation in Biological Swarms

In [1], they explored how a school of shiner fish gains information from the change of position and headings of their neighbors when attacked by a robotic predator fish. In [2, 3], they studied how collective turning emerges from agents fluctuations in natural flocks of starlings. They experimentally validated through real data that turning starts from the agents who are located at the elongated tips of the flock, and then propagates across the remaining agents. In [44], they showed via analyzing real trajectories of Killifish that fish coordinate their position and speed in a way that enriches social communication. In particular, oscillations in speed leads to spatial geometrical configurations such as V formation and diagonal formation. Using a small portion of observed agents, in [45] they developed a Gaussian model based on the mean and variance of the spatial distribution of agents. The behavior of the swarm is represented by the Gaussian distribution over velocity fields. All of these works supports the idea that the spatial mean and variance seen by each agent might are rich in valuable information for analyzing the collective behavior of a swarm.

Various neighboring models describing how agents access information from their neighbors are analyzed in [46, 47]. In particular, the neighbors are selected based on distance in the metric model, based on a number of agents in the topological model, and based on bearing angles in the visual model. Visual models are best for searchings, while topological models are best for avoiding predators. Using the PCA Flow, we can alternatively specify the set of neighbors based on the spatial variances.

In [48], they developed a geometrical framework based on the principal axis of the swarm to analyze the motion of a swarm. The framework results in decomposition of the

motion into kinematic modes such as translations, rotations, expansions, and compressions. This work inspired us to use PCA Flow computed by each agent to detect the change of kinematic modes and use that as a communication signal.

2.4 Consensus-on-a Sphere

Consensus-on-a sphere can be in the form of Kuramoto oscillators that have been treated in [49] for 2D and in [50] for 3D with local stability results. In [51, 21], global convergence results are provided for a generalized consensus on an n-sphere for undirected graphs. Our leader-follower consensus is different in that it deals with directed edges from the followers to the leaders which complicate the analysis. Since the swarm is continuously moving, the leader-follower consensus is time-varying. Through a choice of collective states, we formulate a cascaded input-to-state stability problem, using techniques as in [52]. The collective states allow us to analyze the cascaded input-to-state stability by only requiring connected graphs without any restriction on the graph structure. This is different than for example [53], where the analysis is initially conducted for specific graphs and then inductively generalized to arbitrary graphs.

2.5 Flow Models for Principal Component Analysis

Principal Component Analysis (PCA) is a statistical method that produces directions of maximum variation of a data set [54],[55]. The PCA flow models are dynamical systems in which their solutions converge to the principal components of a given covariance matrix [14]. Among the most used ones are one-unit Oja Flow model [13, 56] and the matrix Xu Flow model [57]. The Oja Flow model converges to the eigenvector corresponding to the largest eigenvalue of the covariance matrix. On the other hand, the Xu Flow model converges to the matrix of all the eigenvectors. Both of them require the covariance matrix to positive definite. In [58], this requirement is relaxed to include positive semi-definite covariance matrices by slightly modifying the flow. In this Dissertation, we used the one-

unit Oja Flow model for the swarming applications in a 2-dimensional space. We will use the Xu Flow model in the proposed work for swarming in a 3-dimensional space. To the best of our knowledge, we are not aware of any existing use of PCA flow models in control of multi-agent systems.

Global convergence analysis of the Oja Flow is given in [56]. Different Lyapunov based convergence results for Oja and Xu flow models is developed in [15, 59]. In this Dissertation, we developed our own Lyapunov function for the Oja flow model that is more appropriate to the setup of our problem and used it to obtain an exponential convergence result.

2.6 Local Information for Multi-Agent Systems

In the recent paper [5], the authors surveyed the locally available information used for the multi-agent formation control and swarming problems. Majority of the existing techniques are position-based, displacement-based, distance-based and angle-based. In the displacement-based, each agent needs to sense the relative position of its neighbor with respect to its local body frame. Additionally, each agent needs to sense the orientation of its body frame with respect to the world frame which can practically be obtained using compass devices. Displacement-based methods require less sensing capabilities than the position-based, and at the same time require fewer interactions with neighbors than the distance-based. The PCA-based body frame requires each agent to sense the relative position, however, it does not require the orientation of the body frame with respect to the world frame.

Very recently, the authors in [4] review the existing multi-agent algorithms for collective behaviors. In particular, they compare various algorithms based on the level of scalability and the amount of required communication bandwidth. Our proposed control paradigm is scalable to an arbitrary number of agents. Additionally, our strategy requires zero communication bandwidth; no zero bandwidth strategy is reported in [4].

CHAPTER 3
USING CONSENSUS-ON-A SPHERE FOR A GRADIENT-FREE DISTRIBUTED SOURCE SEEKING STRATEGY

3.1 Introduction

Distributed source seeking is a collective behavior that has been studied in natural swarms such as flocks of bird and schools of fish, where the source value represents an environmental characteristic such as chemical concentration, light intensity or temperature, just to name a few [6]. Due to its distributed nature, this collective behavior has inspired researchers to develop strategies and algorithms for swarms of robots that are required to localize and identify a feature of interest. The majority of these source seeking strategies require either exact knowledge or explicit estimation of the gradient, which relies on the exchange of field measurements as is the case of [27, 28, 29, 26]. Recently, a 2D bio-inspired distributed source seeking strategy, called the Speeding Up and Slowing Down (SUSD) strategy has been developed in [30] where agents do not need to share their field measurements, but still move towards the source collectively.

In this chapter, we extend the 2D SUSD strategy to the 3D setting for an M-agent swarms. Motivated by certain biological swarming behaviors [60, 61], we consider three agents of the swarm to locally compute a 3D time-varying moving frame based only on relative positions. On the other hand, the rest of the agents compute their body frame using a nonlinear consensus-on-a sphere. Biologically the three agents can be viewed as agents with older ages or more experience, which from a robotic point of view can be thought of as robots with high sensing and computation capabilities. Each agent decomposes its velocity into forward motion in one direction of the frame (SUSD direction), and formation or connectivity-maintaining motion in the plane formed by the remaining components of

the frame (formation plain). The forward motion speed (SUSD speed) depends only on the current field value measured by each agent, hence each agent speeds up or slows down as the field value changes.

The strategy results in a two-layer system composed of environmental and social layers. In the environmental layer, the agents interact with the environment by modulating their speed as a function of the field value. In the social layer, the agents interact with each other by means of implicit and explicit consensus and formation control laws. The implicit consensus law between the SUSD direction and the negative direction of the gradient remarkably emerges from the local interaction rules that the three agents apply to compute their body frame components. The explicit consensus is used by the remaining agents to align with the three agents. This results in a global synchronization behavior where the synchronized value is indirectly controlled by the field value in the environmental layer. This phenomenon can not be explained by known consensus algorithms because the consensus emerges not due to sharing the values or gradient of the field, but due to the SUSD strategy.

A difficulty we overcome in this chapter is to pursue source seeking in a 3D space without assuming knowledge of the gradient as in [26], and without explicit gradient estimation that relies on the exchange of measurements as in [27, 28, 29]. The proposed SUSD strategy assumes that each agent is able to measure the field value only at its current position and does not require agents to exchange field measurements. Although extremum-based swarm source seeking approaches do not require explicit knowledge of the field gradient, they are designed to indirectly estimate the field gradient. Additionally, multi-agent extremum seeking approaches require an exchange of the field measurements [32, 62]. Except for the three agents that are required to maintain an equilateral rigid body, the remaining agents are not required to form any specific formations or graph structure, as in [27] where a circular formation is required. This implies that the strategy is scalable to swarms of arbitrary numbers of agents and connected graphs.

Another challenge we overcome is the convergence and robustness analysis of the nonlinear consensus-on-a sphere control laws. Although the general form of consensus-on-a-sphere is previously analyzed in [51, 21], our analysis is different in that it deals with directed edges that complicate the analysis. Furthermore, since the swarm is continuously moving, the consensus considered in this chapter is time-varying with an input disturbance due to the change of the field gradient as the swarm navigates. Through a choice of collective states that represent the desired source seeking behavior, we are able to lump all the individual consensus states into an overall system of two cascaded subsystems, and then justify the robustness of the consensus law by formulating a cascaded input-to-state stability problem, using techniques as in [52]. The choice of the collective states allows us to analyze the cascaded input-to-state stability by only requiring connected graphs without any restriction on the graph structure. This is different than for example [53], where the analysis is initially conducted for specific graphs and then generalized to arbitrary graphs inductively.

The main contributions of this chapter are (1) extending the SUSD in [30] from 2D to 3D for three-agents formation, (2) integrating a consensus-on-a sphere control law with the SUSD for a swarm of an arbitrary number of agents, (3) proving the convergence and robustness of the proposed strategy through an input-to-state stability analysis [8, 9].

The rest of this chapter is organized as follows. We first formulate the source seeking problem in Section 3.2. Then we design the velocity control law in Section 3.3. After that, the convergence analysis of the proposed strategy to the source location is presented in Section 3.4. Finally simulation results and concluding remarks are given in Section 3.5 and Section 3.6, respectively.

3.2 Problem Formulation

Consider a swarm of M agents in a 3D space. Let $\mathbf{r}_i \in \mathbb{R}^3, i = 1, \cdots, M$ be the position of the i^{th} agent in the 3D space. Let the interaction among the agents be described by a

graph $\mathcal{G} \subseteq \mathcal{V} \times \mathcal{E}$, where $\mathcal{V}$ is the set of all agents and $\mathcal{E}$ is the set of all edges. Additionally, an edge $(i, j) \in \mathcal{E}$ is undirected if also $(j, i) \in \mathcal{E}$ where $i, j \in \mathcal{V}$. Consequently, a graph is undirected if all edges in $\mathcal{E}$ are undirected. A graph is connected if, for all pairs of the agents in the graph, there exists a path connecting the two agents. A graph is complete if each agent shares an edge with all other agents. The neighbor set of i is defined by $\mathcal{N}_i = \{j | (i, j) \in \mathcal{E}\}$. Additionally, if for each agent $\mathcal{N}_i$ is fixed, the graph is static, otherwise it is dynamic. In this chapter, we assume

Assumption 3.2.1. *$\mathcal{G}$ is static and connected.*

Assumption 3.2.2. *Each agent i, is able to measure the relative displacement $(\mathbf{r}_j - \mathbf{r}_i)$ for all $j \in \mathcal{N}_i$.*

In practice, robots can be equipped with sensors to measure the relative positions of their neighbors, which is less challenging than requiring the global positions [5].

Suppose each agent is able to measure a positive field value $z_i = z(\mathbf{r}_i) \in \mathbb{R}$ that represents an environmental characteristic such as temperature or light intensity, with the following assumption

Assumption 3.2.3. *1. The field $z(\mathbf{r}_i)$ is smooth, time-invariant and bounded, i.e. $0 \leq z_{min} \leq z(\mathbf{r}_i) \leq z_{max}$.*

2. The field has a unique minimum at the source location $\mathbf{r}_0$, i.e. $z(\mathbf{r}_0) = z_{min}$.

Although not all of the real fields are smooth [63], this assumption does not limit the applicability of the proposed strategy. For non-smooth fields, we can use stochastic models to transfer them into smooth fields. Indeed, in our preliminary work in [31], we use a Poisson counting process to transform a turbulent plume field into a smooth field.

Let the velocity of each agent be described by

$$\dot{\mathbf{r}}_i = \mathbf{v}_i. \tag{3.1}$$

Then the problem we want to solve is to design a velocity controller $\mathbf{v}_i$ such that the swarm converges to the source distributively. The problem is challenging since we want to solve it without explicitly estimating the field gradient and without sharing field measurements. This design might be used to model a biological swarm of a school of fish seeking dark areas [16], which presumably do not explicitly share field measurements. Additionally, this problem is important for swarm robotics with limited resources such as underwater robotics, and in environments where a field gradient is not possible to be estimated or not well-defined such as a turbulent field.

3.3 Velocity Control Law Design

In this section, we first design source seeking control laws for a 3-agent swarm. We then design control laws that enable a swarm of an arbitrary number of agents to get to the source distributively.

3.3.1 A Swarm of Three Agents

Let the three agents form an equilateral formation, which means that $||\mathbf{r}_1-\mathbf{r}_2|| = ||\mathbf{r}_2-\mathbf{r}_3|| = ||\mathbf{r}_3-\mathbf{r}_1||$. Define a right-handed orthonormal frame $(\mathbf{q},\mathbf{p},\mathbf{n})$, with an origin located at the center $\mathbf{r}_c = \frac{1}{3}\sum_{i=1}^{3}\mathbf{r}_i$, as

$$\mathbf{q} = \frac{\mathbf{r}_2-\mathbf{r}_1}{||\mathbf{r}_2-\mathbf{r}_1||},\ \mathbf{p} = \frac{\mathbf{r}_3-\mathbf{r}_c}{||\mathbf{r}_3-\mathbf{r}_c||},\ \mathbf{n} = \mathbf{q}\times\mathbf{p}. \tag{3.2}$$

Fig. 3.1 illustrates the geometry of the three-agent group with the defined frame where $\mathbf{N} = \frac{\nabla z(\mathbf{r}_c)}{||\nabla z(\mathbf{r}_c)||}$ is a unit length vector pointing towards the field gradient at the center of the formation.

Assumption 3.3.1. *The three agents are numbered from 1 to 3 such that each agent knows the numbers of the other agents.*

Without explicit communication, this assumption can be satisfied by assigning each

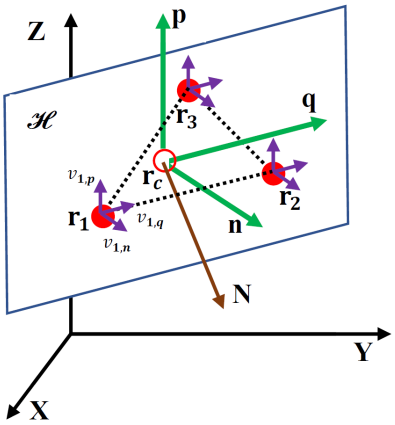


Figure 3.1: Geometry of a three-agent group in located at $\mathbf{r}_1$, $\mathbf{r}_2$, and $\mathbf{r}_3$ at the inertial frame $(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$, where $(\mathbf{n}, \mathbf{p}, \mathbf{q})$ are the body frame components and $\mathbf{N}$ is the gradient direction. The plane $\mathcal{H}$ is formed by the span of the vectors $\mathbf{q}$ and $\mathbf{p}$.

agent with a specific mark or a blinking LED that can be identified by each robot. Given the defined frame, let the velocity $\mathbf{v}_i$ in (4.1) be decomposed as

$$\mathbf{v}_i = \mathbf{v}_{i,\mathbf{n}} + \mathbf{v}_{i,\mathbf{q}} + \mathbf{v}_{i,\mathbf{p}} = v_{i,\mathbf{n}}\mathbf{n} + v_{i,\mathbf{q}}\mathbf{q} + v_{i,\mathbf{p}}\mathbf{p}, \tag{3.3}$$

where $v_{i,\mathbf{n}}$, $v_{i,\mathbf{q}}$ and $v_{i,\mathbf{p}}$ represent the decoupled forward speed and formation speeds along the $\mathbf{n}$, $\mathbf{q}$ and $\mathbf{p}$ directions respectively. Through this decomposition, we decouple the normal component of the velocity $\mathbf{v}_{i,\mathbf{n}}$ from the tangential components $\mathbf{v}_{i,\mathbf{q}}$ and $\mathbf{v}_{i,\mathbf{p}}$, which allows us to analyze the stability of the normal and tangential modes separately.

The objective for $v_{i,\mathbf{q}}$ and $v_{i,\mathbf{p}}$ is to maintain a rigid formation of an equilateral triangle in the plane $\mathcal{H}$ which is formed by the vectors $\mathbf{q}$ and $\mathbf{p}$. Let r_i^q be the projection of $\mathbf{r}_i$ onto vector $\mathbf{q}$, and r_i^p be the projection of $\mathbf{r}_i$ onto vector $\mathbf{p}$, as illustrated in Fig. 3.2 (a) and (b), respectively. For agent i, we define sets $\mathcal{N}_i^q$ and $\mathcal{N}_i^p$ that contain the indices of the neighboring agents along directions $\mathbf{q}$ and $\mathbf{p}$, respectively. For example, for the three-agent group as shown in Fig. 3.2, $\mathcal{N}_1^q = \{3\}$, $\mathcal{N}_2^q = \{3\}$, $\mathcal{N}_3^q = \{1, 2\}$, $\mathcal{N}_1^p = \{2, 3\}$, $\mathcal{N}_2^p = \{1, 3\}$, $\mathcal{N}_3^p = \{1, 2\}$. The goal is to design $v_{i,\mathbf{q}}$ and $v_{i,\mathbf{p}}$ so that the relative distance from r_i^q to $r_j^q, i \neq j$, converges to a constant a_{ij}^0, and the relative distance from r_i^p to

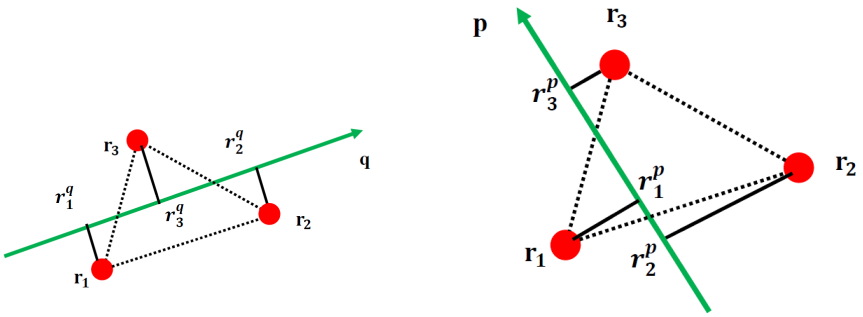


Figure 3.2: Projections of the positions $\mathbf{r}_i$ onto the frame components $\mathbf{q}$ (a) and $\mathbf{p}$ (b).

$r_j^p, i \neq j$, converges to a constant b_{ij}^0. Therefore, we design

$$v_{i,\mathbf{q}} = k_3 \sum_{j \in \mathcal{N}_i^q} [\langle \mathbf{r}_j - \mathbf{r}_i, \mathbf{q} \rangle - a_{j,i}^0], \tag{3.4}$$

$$v_{i,\mathbf{p}} = k_4 \sum_{j \in \mathcal{N}_i^p} [\langle \mathbf{r}_j - \mathbf{r}_i, \mathbf{p} \rangle - b_{j,i}^0], \tag{3.5}$$

where $k_3, k_4 > 0$ are formation gain constants, $a_{i,j}^0 = -a_{j,i}^0$ and $b_{i,j}^0 = -b_{j,i}^0$ are desired formation distances selected such that the three agents form an equilateral triangle.

Inspired by behaviors of fish schools[16], we design the forward speed $v_{i,\mathbf{n}}$ in the direction $\mathbf{n}$ to be proportional to the field value $z(\mathbf{r}_i)$ as follows

$$v_{i,\mathbf{n}} = k_1 z(\mathbf{r}_i) + k_2, \ i = 1, \cdots, M, \tag{3.6}$$

where $k_1, k_2 \in \mathbf{R}$ are positive gain constants. Note that $v_{i,\mathbf{n}}$ depends only on the locally measured field value, $z(\mathbf{r}_i)$. Thus, the forward motion speed increases or decreases based on the field measurement, $z(\mathbf{r}_i)$, and hence it is called Speeding Up and Slowing Down (SUSD) speed. From now and after, we call $v_{i,\mathbf{n}}$ the SUSD speed, and $\mathbf{n}$ the SUSD direction.

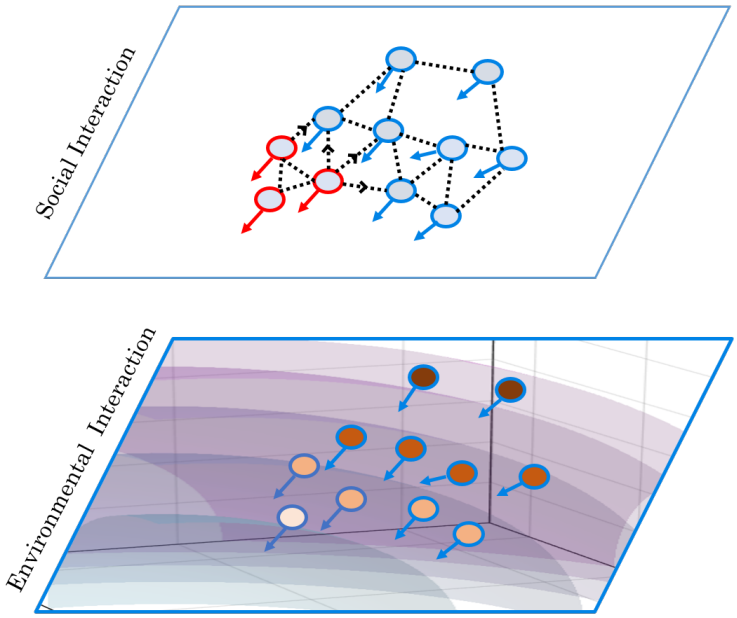


Figure 3.3: At the Environmental layer, each agent, colored by the environmental value, individually modulate its speed according to the instantaneous environmental value. On the other hand, agents coordinate their motion directions on the Social layer based on an implicit consensus law for the red agents and based on an explicit consensus law for the blue agent.

3.3.2 A Swarm of More Than Three Agents

To enable a swarm of more than three agents to reach the source distributively, it is challenging to define the frame components $(\mathbf{q}, \mathbf{p}, \mathbf{n})$ based only on relative positions as in (3.2). Inspired by biological swarms where some agents are assumed to be more capable or more experienced [60, 64], we require three agents to define their SUSD direction (forward direction) as in (3.2), while the rest determine their SUSD direction using the following consensus law

$$\dot{\mathbf{n}}_i = k_f \sum_{j \in \mathcal{N}_i} [\mathbf{n}_j - \langle \mathbf{n}_i, \mathbf{n}_j \rangle \mathbf{n}_i], \; i = 4, \cdots, M, \tag{3.7}$$

where $k_f \in \mathbf{R}$ is a positive constant representing the consensus gain. This is a time-varying nonlinear consensus-on-a sphere which preserves a unit length of its vectors $\mathbf{n}_i$.

Define $\mathcal{G}^l = (\mathcal{V}^l, \mathcal{E}^{ll})$ to be the complete undirected graph describing the interactions

among agents $i = 1, 2, 3$. Similarly, define $\mathcal{G}^f = (\mathcal{V}^f, \mathcal{E}^{ff})$, to be the undirected graph describing interactions among agents $i = 4, \cdots, M$. Additionally, define $\mathcal{E}^{fl}$ to be the set of all edges that have $i \in \mathcal{V}^f$ as a tail and $j \in \mathcal{V}^l$ as a head. According to Assumption 3.2.1, we need $\mathcal{G}^f$ to be connected, and additionally we need to ensure that for $t \geq 0$ there exists at least one $i \in \mathcal{V}^f$ such that $(i, j) \in \mathcal{E}^{fl}$. These connectivity requirements are satisfied through formation control laws (3.9) and (3.10). Note that we require the undirected $\mathcal{G}^f$ to be only connected, but with any graph structure. For (3.7) we further assume

Assumption 3.3.2. *Each agent $i \in \mathcal{V}^f$ is able to obtain the SUSD directions $\mathbf{n}_j$, $\forall j \in \mathcal{N}_i$.*

Relying on advanced vision techniques, agents can measure the headings of their neighbors [65]. With some treatment, agents can then satisfy Assumption 3.3.2 by obtaining $\mathbf{n}_j$ of their neighbors from their heading measurements.

Then, the velocity of each agent is decoupled as

$$\mathbf{v}_i = v_{i,\mathbf{n}}\mathbf{n}_i + \mathbf{v}_{i,\mathcal{H}}, \quad i = 1, \cdots, M, \tag{3.8}$$

where the SUSD direction $\mathbf{n}_i$ and the SUSD speeds are as defined by by (3.7) and (3.6), respectively. The formation term $\mathbf{v}_{i,\mathcal{H}}$ is defined to be

$$\mathbf{v}_{i,\mathcal{H}} = v_{i,\mathbf{q}}\mathbf{q} + v_{i,\mathbf{p}}\mathbf{p}, \qquad i = 1, \cdots, 3 \tag{3.9}$$

$$\mathbf{v}_{i,\mathcal{H}} = \sum_{j \in \mathcal{N}_i} w_{ij}(\mathbf{I} - \mathbf{n}_i\mathbf{n}_i^T)(\mathbf{r}_j - \mathbf{r}_i), \qquad i = 4, \cdots, M \tag{3.10}$$

where $\mathbf{q}$ and $\mathbf{p}$ are as defined in (3.2), and $w_{ij} = ||\mathbf{r}_j - \mathbf{r}_i||^2 - \langle \mathbf{r}_j - \mathbf{r}_i, \mathbf{n}_i \rangle^2 - d_{ij}^2$ for an arbitrary desired inter-agent distances d_{ij}. Note that the formation term (3.10) acts only on the plane perpendicular to $\mathbf{n}_i$ so that the SUSD speed along the SUSD direction $\mathbf{n}_i$ is only affected by the field value. Additionally, (3.10) is required to only ensure connectivity maintenance without any specific rigid formations as in (3.9).

The resulted strategy may be viewed as a two-layer system composed of environmental

and social interactions as shown in Fig. 3.3. In the environmental layer, all the agents modulate their SUSD speeds according to the environmental field value as indicated by (3.6). In the social layer, agents interact with each other to determine their SUSD directions as indicated by (3.2) and (3.7), and to maintain formations as indicated by (3.9) and (3.10).

Remark 1. *Although* 3 *agents are enough to seek the source, the proposed strategy presents a new method that enables a swarm of more than* 3 *agents to navigate to the source with local information. This is particularly important in modeling large biological swarms. We discover in Section 3.4 that using the* 3*-agent local interactions (3.2), (3.6), (3.4) and (3.5) leads to a nonlinear consensus-on-a sphere (3.27) between the SUSD direction and the negative direction of the field gradient. This emergent consensus along with the explicit consensus law (3.7) of the rest of the agents in the social layer leads to a synchronization behavior where the synchronized value is indirectly controlled by the field value in the environmental layer of Fig. 3.3. This behavior is not achieved by the classical distance-based leader-follower approaches, where the headings of the followers are always pointing toward the leaders, not the field gradient. Hence, the proposed two-layer model might be more reasonable to describe at least some synchronization behaviors of biological swarms.*

3.4 Convergence Analysis

In [31], given Assumptions 3.2.1, 3.2.2 and 3.2.3, the formation control laws (3.9) is proved to be exponentially stable. When the consensus law (3.7) converges, then all agents will have the same normal plane $\mathcal{H}$, and hence the formation control law (3.10) becomes a known formation problem that can be proved using methods in [31], or others in the related literature. However, during the transient time, each agent will have its own plane $\mathcal{H}_i$, and hence proving the convergence of (3.10) requires more treatment which is beyond the scope of this chapter. Remark that all the following subsequent proofs do not require the convergence of (3.10).

3.4.1 Convergence of the SUSD Direction for $M = 3$

The goal of this section is to show that the SUSD direction $\mathbf{n}$ in (3.2) converges to the negative direction of the gradient, $\mathbf{N} = \frac{\nabla z(\mathbf{r}_c)}{||\nabla z(\mathbf{r}_c)||}$. In the inertial frame, once the formation converges, $v_{i,\mathbf{q}} = v_{i,\mathbf{p}} = 0$. Then, the velocity of the i^{th} agent in the rigid body becomes $\mathbf{v}_i = v_{i,\mathbf{n}}\mathbf{n}$, and the velocity of the formation center is $\mathbf{v}_c = \frac{1}{3}\sum_{i=1}^{3} v_{i,\mathbf{n}}\mathbf{n} = v_{c,\mathbf{n}}\mathbf{n}$, which indicates that the moving direction of the rigid body coincides with the $\mathbf{n}$ axis of the body frame. Define shape variables $\langle \mathbf{N}, \mathbf{n}\rangle$, $\langle \mathbf{N}, \mathbf{q}\rangle$, and $\langle \mathbf{N}, \mathbf{p}\rangle$ [66, 67]. The shape variables satisfy

$$\langle \mathbf{N}, \mathbf{p}\rangle^2 = 1 - \langle \mathbf{N}, \mathbf{q}\rangle^2 - \langle \mathbf{N}, \mathbf{n}\rangle^2. \tag{3.11}$$

Since we have

$$\frac{d\langle \mathbf{N}, \mathbf{n}\rangle}{dt} = \langle \mathbf{N}, \dot{\mathbf{n}}\rangle + \langle \dot{\mathbf{N}}, \mathbf{n}\rangle. \tag{3.12}$$

Then, the first step is to derive $\dot{\mathbf{n}}$. In the frame $(\mathbf{q}, \mathbf{p}, \mathbf{n})$, we can write any vector $\mathbf{v}$ as

$$\mathbf{v} = \langle \mathbf{q}, \mathbf{v}\rangle\mathbf{q} + \langle \mathbf{p}, \mathbf{v}\rangle\mathbf{p} + \langle \mathbf{n}, \mathbf{v}\rangle\mathbf{n}. \tag{3.13}$$

To find $\dot{\mathbf{n}}$, we apply (3.13) with $\mathbf{v} = \dot{\mathbf{n}}$, and calculate the coefficients $\langle \mathbf{q}, \dot{\mathbf{n}}\rangle$, $\langle \mathbf{p}, \dot{\mathbf{n}}\rangle$, and $\langle \mathbf{n}, \dot{\mathbf{n}}\rangle$. In the inertial frame, define the rotation matrix of the rigid body as $\mathbf{g} = [\mathbf{q}, \mathbf{p}, \mathbf{n}] \in SO(3)$. Define a skew symmetric matrix $S(\omega)$, in which $\omega \in \mathbb{R}^3$ is the angular velocity of the rigid body. Then, we have $\dot{\mathbf{g}} = S(\omega)\mathbf{g}$, and from which we derive $\dot{\mathbf{n}} = \omega \times \mathbf{n}$. Since the speed of $\mathbf{r}_i$ along directions $\mathbf{q}$ and $\mathbf{p}$ are zero for the rigid body, we conclude that ω is confined in the plane $\mathcal{H}$. For the velocity of the agent in the inertial frame, $\mathbf{v}_i - \mathbf{v}_c$ satisfies

$$\mathbf{v}_i - \mathbf{v}_c = \omega \times (\mathbf{r}_i - \mathbf{r}_c). \tag{3.14}$$

Then, we have

$$(v_{i,\mathbf{n}} - v_{c,\mathbf{n}})\mathbf{n} = \omega \times (\mathbf{r}_i - \mathbf{r}_c). \tag{3.15}$$

Applying inner product with $\mathbf{n}$ on both sides of (3.15)

$$v_{i,\mathbf{n}} - v_{c,\mathbf{n}} = \langle \omega \times (\mathbf{r}_i - \mathbf{r}_c), \mathbf{n} \rangle. \tag{3.16}$$

Define $\omega_i = \langle \omega \times (\mathbf{r}_i - \mathbf{r}_c), \mathbf{n} \rangle$. We then have

$$\omega_i = -\langle \mathbf{r}_i - \mathbf{r}_c, \omega \times \mathbf{n} \rangle = -\langle \mathbf{r}_i - \mathbf{r}_c, \dot{\mathbf{n}} \rangle. \tag{3.17}$$

Using (3.2) and (3.17), we derive

$$\omega_3 = - \parallel \mathbf{r}_3 - \mathbf{r}_c \parallel \langle \mathbf{p}, \dot{\mathbf{n}} \rangle, \tag{3.18}$$

$$\omega_2 - \omega_1 = - \parallel \mathbf{r}_2 - \mathbf{r}_1 \parallel \langle \mathbf{q}, \dot{\mathbf{n}} \rangle, \tag{3.19}$$

which produces $\langle \mathbf{p}, \dot{\mathbf{n}} \rangle = -\frac{\omega_3}{\|\mathbf{r}_3 - \mathbf{r}_c\|}$ and $\langle \mathbf{q}, \dot{\mathbf{n}} \rangle = -\frac{\omega_2 - \omega_1}{\|\mathbf{r}_2 - \mathbf{r}_1\|}$. Since $\mathbf{n}$ is a unit vector, we have $\langle \mathbf{n}, \dot{\mathbf{n}} \rangle = 0$. Therefore

$$\dot{\mathbf{n}} = -\frac{\omega_2 - \omega_1}{\parallel \mathbf{r}_2 - \mathbf{r}_1 \parallel}\mathbf{q} - \frac{\omega_3}{\parallel \mathbf{r}_3 - \mathbf{r}_c \parallel}\mathbf{p}. \tag{3.20}$$

From (3.16), we have $\omega_i = v_{i,\mathbf{n}} - v_{c,\mathbf{n}}$. Since the field $z(\mathbf{r})$ is at least class C^1, then, from the Taylor expansion, we have

$$v_{i,\mathbf{n}} = k_1(z(\mathbf{r}_c) + \langle \nabla z(\mathbf{r}_c), \mathbf{r}_i - \mathbf{r}_c \rangle) + k_2 + H.O.T, \tag{3.21}$$

where $H.O.T$ represents higher order terms. In addition,

$$\begin{aligned} v_{c,\mathbf{n}} &= \frac{1}{3}\sum_{i=1}^{3} v_{i,\mathbf{n}} \\ &= \frac{k_1}{3}\sum_{i=1}^{3} z(\mathbf{r}_i) + k_2 \\ &= k_1 z(\mathbf{r}_c) + \langle \frac{k_1}{3}\nabla z(\mathbf{r}_c), \sum_{i=1}^{3}\mathbf{r}_i - 3\mathbf{r}_c\rangle + k_2 \\ &= k_1 z(\mathbf{r}_c) + k_2. \end{aligned} \tag{3.22}$$

Therefore, if the agents are close enough to each other such that the higher order terms are insignificant, we derive

$$\omega_i = v_{i,\mathbf{n}} - v_{c,\mathbf{n}} = k_1 \parallel \nabla z(\mathbf{r}_c) \parallel \langle \mathbf{N}, \mathbf{r}_i - \mathbf{r}_c\rangle, \tag{3.23}$$

which leads to

$$\omega_2 - \omega_1 = k_1 \parallel \nabla z(\mathbf{r}_c) \parallel\parallel \mathbf{r}_2 - \mathbf{r}_1 \parallel (\mathbf{N}\cdot\mathbf{q}), \tag{3.24}$$

$$\omega_3 = k_1 \parallel \nabla z(\mathbf{r}_c) \parallel\parallel \mathbf{r}_3 - \mathbf{r}_c \parallel (\mathbf{N}\cdot\mathbf{p}). \tag{3.25}$$

Substituting (3.24) and (3.25) into (3.20), we obtain

$$\dot{\mathbf{n}} = -k_1 \parallel \nabla z(\mathbf{r}_c) \parallel (\langle \mathbf{N}, \mathbf{q}\rangle\mathbf{q} + \langle \mathbf{N}, \mathbf{p}\rangle\mathbf{p}). \tag{3.26}$$

Lemma 3.4.1. *The dynamics (3.26) represents a consensus-on-a sphere control law between the SUSD direction and negative direction of the gradient. In particular, we can*

rewrite (3.26) as

$$\dot{\mathbf{n}} = -k_l||\nabla z(\mathbf{r}_c)||(\mathbf{I} - \mathbf{n}\mathbf{n}^T)\mathbf{N}. \tag{3.27}$$

Proof. From (3.11) we can write

$$\mathbf{N}^T(\mathbf{q}\mathbf{q}^T + \mathbf{p}\mathbf{p}^T)\mathbf{N} = \mathbf{N}^T\mathbf{N} - \mathbf{N}^T\mathbf{n}\mathbf{n}^T\mathbf{N} = \mathbf{N}^T(\mathbf{I} - \mathbf{n}\mathbf{n}^T)\mathbf{N}, \tag{3.28}$$

which we use to obtain

$$(\mathbf{q}\mathbf{q}^T + \mathbf{p}\mathbf{p}^T)\mathbf{N} = (\mathbf{I} - \mathbf{n}\mathbf{n}^T)\mathbf{N}. \tag{3.29}$$

Finally, plug (3.29) into (3.26) to get (3.27). □

Note that the consensus (3.27) is a time-varying since $\nabla z(\mathbf{r}_c)$ is changing as the center $\mathbf{r}_c$ moves around. We want to show that the consensus law asymptotically converges to the agreement $\mathbf{n} = -\mathbf{N}$ as $t \to \infty$. Let $\theta = \langle \mathbf{N}, \mathbf{n} \rangle + 1$ and $\delta = \langle \mathbf{n}, \dot{\mathbf{N}} \rangle$. Then using (3.27), we derive

$$\begin{aligned}
\dot{\theta} &= \langle \mathbf{N}, \dot{\mathbf{n}} \rangle + \langle \mathbf{n}, \dot{\mathbf{N}} \rangle \\
&= -k_l \parallel \nabla z(\mathbf{r}_c) \parallel (1 - \langle \mathbf{N}, \mathbf{n} \rangle^2) + \langle \mathbf{n}, \dot{\mathbf{N}} \rangle \\
&= -k_l||\nabla z(\mathbf{r}_c)||\theta(2 - \theta) + \delta \\
&\triangleq h(t, \theta, \delta).
\end{aligned} \tag{3.30}$$

The unforced system $h(t, \theta, 0)$, has two equilibriums: $\theta = 0$ and $\theta = 2$, where $\theta = 0$ corresponds to the desired equilibrium $\langle \mathbf{N}, \mathbf{n} \rangle = -1$, and $\theta = 2$ corresponds to the undesired equilibrium $\langle \mathbf{N}, \mathbf{n} \rangle = 1$. Since we don't know $\dot{\mathbf{N}}$, we view δ as an input disturbance and analyze the system convergence using an input-to-state stability framework. Note that since $\mathbf{N}$ is perpendicular to $\dot{\mathbf{N}}$, then $\delta = 0$ when $\theta = 0, 2$. Theorem 3.4.1 summarizes the

stability results of (3.30).

Theorem 3.4.1. *Consider (3.30). Assume that* $||\nabla z(\mathbf{r}_c)||$ *is bounded below along the trajectory of the formation center, i.e.* $||\nabla z(\mathbf{r}_c)|| \geq \epsilon_c$ *for a small constant* $\epsilon_c > 0$ *everywhere except at the source location where* $z(\mathbf{r}_c) = 0$*. If initially* $\theta(0) \neq 2$*, then the equilibrium* $\theta = 0$ *of the unforced system,* $h(t, \theta, 0)$ *is asymptotically stable. Moreover, whenever* $\theta(0) \neq 2$ *and assuming* $|\delta| < 2k\epsilon\epsilon_c$ *for a small* $\epsilon < 1$*, system (3.30) is input-to-state stable.*

Proof. Let $\mathbf{D} = \{\theta \in \mathbb{R} | 0 \leq \theta < 2\}$. Let $V(\theta) : \mathbf{D} \to \mathbf{R}$ be a Lyapunov candidate function defined as follows

$$V(\theta) = \frac{\theta}{2 - \theta}. \tag{3.31}$$

Note that $V \geq 0$ and $V = 0$ if and only if $\theta = 0$. Furthermore, $V \to \infty$ as $\theta \to 2$. Then

$$\dot{V} = \frac{\partial V}{\partial \theta}\dot{\theta} = \frac{2}{(2 - \theta)^2}\dot{\theta}. \tag{3.32}$$

For the unforced system $h(t, \theta, 0)$, we have

$$\dot{V} = \frac{-2k||\nabla z(\mathbf{r}_c)||\theta(2 - \theta)}{(2 - \theta)^2} = -2k||\nabla z(\mathbf{r}_c)||V \leq 0. \tag{3.33}$$

Since $V \geq 0$ and $V = 0$ if and only if $\theta = 0$, then the equilibrium $\theta = 0$ of the unforced system is asymptotically stable. Moreover, since $\dot{V}$ is negative definite and $V \to \infty$ whenever $\theta \to 2$, then $\mathbf{D} = \{\theta \in \mathbb{R} | 0 \leq \theta < 2\}$ is a positively invariant set which implies that trajectories start inside $\mathbf{D}$ will stay there forever. For the forced system $h(t, \theta, \delta)$

$$\begin{aligned} \dot{V} &= \frac{-2k||\nabla z(\mathbf{r}_c)||\theta(2 - \theta)}{(2 - \theta)^2} + \frac{2\delta}{(2 - \theta)^2} \\ &\leq -2k||\nabla z(\mathbf{r}_c)||(1 - \epsilon)V, \quad \forall |\theta| > \frac{|\delta|}{2k\epsilon\epsilon_c}. \end{aligned} \tag{3.34}$$

Let $\alpha_1(|\theta|) = \alpha_2(|\theta|) = \frac{|\theta|}{2 - |\theta|}$, which are class $\mathcal{K}_\infty$ functions on $\mathbf{D}$ and satisfy: $\alpha_1(|\theta|) \leq$

$V(\theta) \leq \alpha_2(|\theta|)$. Additionally, $\alpha_3(\theta) = 2k||\nabla z(\mathbf{r}_c)||(1-\epsilon)\frac{\theta}{2-\theta}$ and $\rho(|\delta|) = \frac{|\delta|}{2k\epsilon\epsilon_c}$ are class $\mathcal{K}$ functions. Therefore, according to *Theorem 4.19* in [68], the system (3.30) is input-to-state stable with gain $\gamma = \alpha_1^{-1} \circ \alpha_2 \circ \rho = \frac{|\delta|}{2k\epsilon\epsilon_c}$. Additionally, since δ is vanishing at the equilibrium, then the system is asymptotically stable. □

3.4.2 Convergence of the SUSD Directions for $M > 3$

The goal of this section is to show that the consensus law (3.7) converges to the solution $\mathbf{n}_i = \mathbf{n}_j = \mathbf{n}\ \forall\ i, j \in \mathcal{V}^f$.

Let $\mathbf{u}_i = \sum_{k\in\mathcal{N}_i} \mathbf{n}_k$. Then rewrite the consensus law (3.7) as

$$\dot{\mathbf{n}}_i = k_f \sum_{k\in\mathcal{N}_i} (\mathbf{n}_k - \mathbf{n}_k^T\mathbf{n}_i\mathbf{n}_i) = k_z(\mathbf{I} - \mathbf{n}_i\mathbf{n}_i^T)\mathbf{u}_i. \tag{3.35}$$

The consensus (3.35) has the following three equilibrium sets

$$(\mathbf{n}_i, \mathbf{u}_i) \in \left\{ \left(-\frac{\mathbf{u}_i}{||\mathbf{u}_i||}, \mathbf{u}_i\right), \left(\frac{\mathbf{u}_i}{||\mathbf{u}_i||}, \mathbf{u}_i\right), (\mathbf{n}_i, \mathbf{0}) \right\} \tag{3.36}$$

As proved in [21], the undesired equilibrium sets $(\mathbf{n}_i, \mathbf{u}_i) = \left(-\frac{\mathbf{u}_i}{||\mathbf{u}_i||}, \mathbf{u}_i\right)$ and $(\mathbf{n}_i, \mathbf{u}_i) = (\mathbf{n}_i, \mathbf{0})$ are unstable. The unique asymptotically stable equilibrium is the set $(\mathbf{n}_i, \mathbf{u}_i) = \left(\frac{\mathbf{u}_i}{||\mathbf{u}_i||}, \mathbf{u}_i\right)$ which is equivalent to $\{\mathbf{n}_i = \mathbf{n}_j = \mathbf{n}_l$ for all $i, j \in \mathcal{V}^f\}$ where by $\mathbf{n}_l$ we denote the SUSD direction of the three agents defined in (3.2).

Since three agents have their own SUSD dynamics given by (3.27), then the analysis of (3.35) is different from the one analyzed in [21]. In particular, in [21] all the edges are undirected whereas in this chapter we need to consider the directed edges $(i, l) \in \mathcal{E}^{fl}$. Additionally, (3.27) is time-varying, which produces a time-varying nonlinear consensus-on-a sphere problem compared to the time-invariant form considered in [21]. To overcome these difficulties, we first construct collective states that represent the desired equilibrium and then derive their dynamics. We then formulate a cascaded input-to-state stability prob-

lem to analyze the stability of (3.35). Consider the following collective states

$$\theta_f = \sum_{(i,j)\in\mathcal{E}^{ff}} \left(1 - \langle \mathbf{n}_i, \mathbf{n}_j \rangle\right), \tag{3.37}$$

$$\theta_l = \sum_{(i,l)\in\mathcal{E}^{fl}} \left(1 - \langle \mathbf{n}_i, \mathbf{n}_l \rangle\right), \tag{3.38}$$

$$\theta_N = 1 + \langle \mathbf{n}_l, \mathbf{N} \rangle, \tag{3.39}$$

where $\theta_f = 0$ if and only if $\mathbf{n}_i = \mathbf{n}_j \ \forall \ (i,j) \in \mathcal{E}^{ff}$, $\theta_l = 0$ if and only if $\mathbf{n}_i = \mathbf{n}_l \ \forall \ (i,l) \in \mathcal{E}^{fl}$, and $\theta_N = 0$ if and only if $\mathbf{n}_l \to \mathbf{N}$ as $t \to \infty$. Hence, the convergence of these collective states, $(\theta_f, \theta_l, \theta_N)$ to the origin, $(0,0,0)$ represents the desired objective of $\mathbf{n}_i \to \mathbf{n}_l \to -\mathbf{N}$ for all agents. The dynamics of the collective state θ_N is given by (3.30). In the following, we derive the dynamics of the collective states θ_f and θ_l. Taking time derivative of (3.37), and using (3.35)

$$\begin{aligned}
\dot{\theta}_f &= - \sum_{(i,j)\in\mathcal{E}^{ff}} \left[\langle \mathbf{n}_i, \dot{\mathbf{n}}_j \rangle + \langle \mathbf{n}_j, \dot{\mathbf{n}}_i \rangle\right] \\
&= -k_f \sum_{(i,j)\in\mathcal{E}^{ff}} \left[\langle \mathbf{n}_i, (\mathbf{I} - \mathbf{n}_j \mathbf{n}_j^T)\mathbf{u}_j \rangle + \langle \mathbf{n}_j, (\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T)\mathbf{u}_i \rangle\right].
\end{aligned}$$

To continue, we first prove the following *Lemma*

Lemma 3.4.2. *For a connected and undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$,*

$$\sum_{(i,j)\in\mathcal{E}} \left[\langle \mathbf{n}_i, (\mathbf{I} - \mathbf{n}_j \mathbf{n}_j^T)\mathbf{u}_j \rangle + \langle \mathbf{n}_j, (\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T)\mathbf{u}_i \rangle\right] = \sum_{i\in\mathcal{V}} \sum_{j\in\mathcal{N}_i} \langle \mathbf{n}_j, (\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T)\mathbf{u}_i \rangle. \tag{3.40}$$

Proof. Since (i,j) is an undirected edge, then

$$\begin{aligned}
&\sum_{(i,j)\in\mathcal{E}} \left[\langle \mathbf{n}_i, (\mathbf{I}-\mathbf{n}_j\mathbf{n}_j^T)\mathbf{u}_j\rangle + \langle \mathbf{n}_j, (\mathbf{I}-\mathbf{n}_i\mathbf{n}_i^T)\mathbf{u}_i\rangle\right] \\
&= 2\sum_{(i,j)\in\mathcal{E}} \langle \mathbf{n}_j, (\mathbf{I}-\mathbf{n}_i\mathbf{n}_i^T)\mathbf{u}_i\rangle \\
&= 2(\frac{1}{2})\sum_{i\in\mathcal{V}}\sum_{j\in\mathcal{N}_i} \langle \mathbf{n}_j, (\mathbf{I}-\mathbf{n}_i\mathbf{n}_i^T)\mathbf{u}_i\rangle \\
&= \sum_{i\in\mathcal{V}}\sum_{j\in\mathcal{N}_i} \langle \mathbf{n}_j, (\mathbf{I}-\mathbf{n}_i\mathbf{n}_i^T)\mathbf{u}_i\rangle. \qquad (3.41)
\end{aligned}$$

□

Note that, the Lemma requires (i,j) to be an undirected edge. The fact that some directed edges appear under $\mathbf{u}_i = \sum_{k\in\mathcal{N}_i}\mathbf{n}_k$ or $\mathbf{u}_j = \sum_{k\in\mathcal{N}_j}\mathbf{n}_k$ for some i,j does not violate the lemma, since they are still captured under $\mathbf{u}_i$. Since $\mathcal{E}^{ff}$ is the set of all undirected edges, we use **Lemma** 3.4.2 to get

$$\begin{aligned}
\dot{\theta}_f &= -k_f \sum_{i\in\mathcal{V}^f}\sum_{j\in\mathcal{N}_i} \langle \mathbf{n}_j, (\mathbf{I}-\mathbf{n}_i\mathbf{n}_i^T)\mathbf{u}_i\rangle \\
&= -k_f \sum_{i\in\mathcal{V}^f} \langle \sum_{j\in\mathcal{N}_i} \mathbf{n}_j, (\mathbf{I}-\mathbf{n}_i\mathbf{n}_i^T)\mathbf{u}_i\rangle. \qquad (3.42)
\end{aligned}$$

Note that $\sum_{j\in\mathcal{N}_i}\mathbf{n}_j \neq \mathbf{u}_i = \sum_{k\in\mathcal{N}_i}\mathbf{n}_k$ since $k\in\{1,2,3,\cdots,M\}$, while $j\in\{4,\cdots,M\}$. For this, we write

$$\mathbf{u}_i = a_i\mathbf{n}_l + \sum_{j\in\mathcal{N}_i}\mathbf{n}_j = a_i\mathbf{n}_l + \hat{\mathbf{u}}_i, \qquad (3.43)$$

where $a_i = 1$, if one of the agents in the set $\{1,2,3\}$ is a neighbor to the i^{th} agent, and

$a_i = 0$, otherwise. Then we continue to obtain

$$\begin{aligned}
\dot{\theta}_f &= -k_f \sum_{i \in \mathcal{V}^f} \langle \hat{\mathbf{u}}_i, (\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T)(a_i \mathbf{n}_l + \hat{\mathbf{u}}_i) \rangle \\
&= k_f \left[\sum_{i \in \mathcal{V}^f} \left(\langle \hat{\mathbf{u}}_i, \mathbf{n}_i \rangle^2 - ||\hat{\mathbf{u}}_i||^2 \right) - \sum_{i \in \mathcal{V}^f} \langle \hat{\mathbf{u}}_i, a_i (\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T) \mathbf{n}_l \rangle \right] \\
&= k_f \left[\sum_{i \in \mathcal{V}^f} \left(\langle \hat{\mathbf{u}}_i, \mathbf{n}_i \rangle^2 - ||\hat{\mathbf{u}}_i||^2 \right) - \sum_{(i,l) \in \mathcal{E}^{fl}} \langle \hat{\mathbf{u}}_i, (\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T) \mathbf{n}_l \rangle \right]. \qquad (3.44)
\end{aligned}$$

Similarly, taking time derivative of (3.38), and using (3.35) and (3.27)

$$\begin{aligned}
\dot{\theta}_l &= - \sum_{(i,l) \in \mathcal{E}^{fl}} \left[\langle \mathbf{n}_l, \dot{\mathbf{n}}_i \rangle + \langle \mathbf{n}_i, \dot{\mathbf{n}}_l \rangle \right] \\
&= -k_f \sum_{(i,l) \in \mathcal{E}^{fl}} \langle \mathbf{n}_l, (\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T) \mathbf{u}_i \rangle + k_l ||\nabla z(\mathbf{r}_c)|| \sum_{(i,l) \in \mathcal{E}^{fl}} \langle \mathbf{n}_i, (\mathbf{I} - \mathbf{n}_l \mathbf{n}_l^T) \mathbf{N} \rangle \\
&= -k_f \sum_{(i,l) \in \mathcal{E}^{fl}} \langle \mathbf{n}_l, (\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T)(\hat{\mathbf{u}}_i + \mathbf{n}_l) \rangle + k_l ||\nabla z(\mathbf{r}_c)|| \sum_{(i,l) \in \mathcal{E}^{fl}} \langle \mathbf{n}_i, (\mathbf{I} - \mathbf{n}_l \mathbf{n}_l^T) \mathbf{N} \rangle \\
&= k_f \sum_{(i,l) \in \mathcal{E}^{fl}} (\langle \mathbf{n}_i, \mathbf{n}_l \rangle^2 - 1) - k_f \sum_{(i,L) \in \mathcal{E}^{fl}} \langle \hat{\mathbf{u}}_i, (\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T) \mathbf{n}_l \rangle \\
&+ k_l ||\nabla z(\mathbf{r}_c)|| \sum_{(i,l) \in \mathcal{E}^{fl}} \langle \mathbf{n}_i, (\mathbf{I} - \mathbf{n}_l \mathbf{n}_l^T) \mathbf{N} \rangle. \qquad (3.45)
\end{aligned}$$

Let $x_1 = \theta_f + \theta_l$ and $x_2 = \theta_N$. We then view the system as a cascade of two systems

$$\begin{aligned}
\dot{x}_1 &= f_1(x_1, x_2), \\
\dot{x}_2 &= f_2(x_2, \delta). \qquad (3.46)
\end{aligned}$$

Note that the f_1 system represents the consensus among all SUSD directions of the agents in the set $\{4, \cdots, M\}$ with x_2 represents the input disturbance due to the dynamics of the agents in the set $\{1, 2, 3\}$. On the other hand, the f_2 system represents the consensus of the SUSD direction of the agents in the set $\{1, 2, 3\}$ with the negative direction of the gradient in which δ represents the input disturbance due to the dynamics of the gradient. Using,

(3.44) and (3.45), we derive the dynamics of x_1

$$\begin{aligned}\dot{x}_1 &= \dot{\theta}_f + \dot{\theta}_l \\ &= k_f \sum_{i \in \mathcal{V}^f} \left(\langle \hat{\mathbf{u}}_i, \mathbf{n}_i \rangle^2 - ||\hat{\mathbf{u}}_i||^2 \right) + k_f \sum_{(i,l) \in \mathcal{E}^{fl}} \left(\langle \mathbf{n}_i, \mathbf{n}_l \rangle^2 - 1 \right) \\ &- 2k_f \sum_{(i,l) \in \mathcal{E}^{fl}} \langle \hat{\mathbf{u}}_i, (\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T) \mathbf{n}_l \rangle + k_l ||\nabla z(\mathbf{r}_c)|| \sum_{(i,l) \in \mathcal{E}^{fl}} \langle \mathbf{n}_i, (\mathbf{I} - \mathbf{n}_l \mathbf{n}_l^T) \mathbf{N} \rangle. \end{aligned} \tag{3.47}$$

To further simplify, we utilize the following lemma

Lemma 3.4.3.

$$\begin{aligned}-2\langle \hat{\mathbf{u}}_i, (\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T) \mathbf{n}_l \rangle = \\ \Big[\langle \mathbf{n}_i, \mathbf{u}_i \rangle^2 - ||\mathbf{u}_i||^2] + [||\hat{\mathbf{u}}_i||^2 - \langle \hat{\mathbf{u}}_i, \mathbf{n}_i \rangle^2] + [1 - \langle \mathbf{n}_i, \mathbf{n}_l \rangle^2 \Big].\end{aligned} \tag{3.48}$$

Proof.

$$\begin{aligned}-2\langle \hat{\mathbf{u}}_i, (\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T) \mathbf{n}_l \rangle &= 2\langle \hat{\mathbf{u}}_i, \mathbf{n}_i \rangle \langle \mathbf{n}_i, \mathbf{n}_l \rangle - 2\langle \hat{\mathbf{u}}_i, \mathbf{n}_l \rangle \\ &= \left(\langle \hat{\mathbf{u}}_i, \mathbf{n}_i \rangle + \langle \mathbf{n}_i, \mathbf{n}_l \rangle \right)^2 - \langle \hat{\mathbf{u}}_i, \mathbf{n}_i \rangle^2 - \langle \mathbf{n}_i, \mathbf{n}_l \rangle^2 \\ &- \langle \hat{\mathbf{u}}_i + \mathbf{n}_l, \hat{\mathbf{u}}_i + \mathbf{n}_l \rangle + ||\hat{\mathbf{u}}_i||^2 + 1 \\ &= \langle \mathbf{n}_i, \hat{\mathbf{u}}_i + \mathbf{n}_l \rangle^2 - ||\hat{\mathbf{u}}_i + \mathbf{n}_l||^2 + ||\hat{\mathbf{u}}_i||^2 - \langle \hat{\mathbf{u}}_i, \mathbf{n}_i \rangle^2 + 1 - \langle \mathbf{n}_i, \mathbf{n}_l \rangle^2.\end{aligned} \tag{3.49}$$

Since $\mathbf{u}_i = \hat{\mathbf{u}}_i + \mathbf{n}_l \; \forall \; (i,l) \in \mathcal{E}^{fl}$, then the lemma follows directly from the last step. $\square$

Therefore, applying **Lemma** 3.4.3 in (3.47)

$$\begin{aligned}\dot{x}_1 &= k_f \left[\sum_{i\in\mathcal{V}^f} \left(\langle \hat{\mathbf{u}}_i, \mathbf{n}_i \rangle^2 - ||\hat{\mathbf{u}}_i||^2 \right) + \sum_{(i,l)\in\mathcal{E}^{fl}} \left(||\hat{\mathbf{u}}_i||^2 - \langle \hat{\mathbf{u}}_i, \mathbf{n}_i \rangle^2 \right) \right. \\ &\quad \left. + \sum_{(i,l)\in\mathcal{E}^{fl}} \left(\langle \mathbf{n}_i, \mathbf{u}_i \rangle^2 - ||\mathbf{u}_i||^2 \right) \right] + k_l ||\nabla z(\mathbf{r}_c)|| \sum_{(i,l)\in\mathcal{E}^{fl}} \langle \mathbf{n}_i, (\mathbf{I} - \mathbf{n}_l \mathbf{n}_l^T)\mathbf{N} \rangle \\ &= k_f \left[\sum_{i\in\tilde{\mathcal{V}}^f} \left(\langle \hat{\mathbf{u}}_i, \mathbf{n}_i \rangle^2 - ||\hat{\mathbf{u}}_i||^2 \right) + \sum_{(i,l)\in\mathcal{E}^{fl}} \left(\langle \mathbf{n}_i, \mathbf{u}_i \rangle^2 - ||\mathbf{u}_i||^2 \right) \right] \\ &\quad + k_l ||\nabla z(\mathbf{r}_c)|| \sum_{(i,l)\in\mathcal{E}^{fl}} \langle \mathbf{n}_i, (\mathbf{I} - \mathbf{n}_l \mathbf{n}_l^T)\mathbf{N} \rangle, \end{aligned} \tag{3.50}$$

where $\tilde{\mathcal{V}}^f = \mathcal{V}^f - \{i | (i,l) \in \mathcal{E}^{fl}\}$. The following theorem summarizes the convergence results of the cascaded system (3.46).

Theorem 3.4.2. *Consider (3.46) with $f_1(x_1, x_2)$ and $f_2(x_2, \delta)$ are as defined in (3.50) and (3.30), respectively. Assume that $||\nabla z(\mathbf{r}_c)||$ is bounded below along the trajectory of agents $\{1, 2, 3\}$, i.e. $||\nabla z(\mathbf{r}_c)|| \geq \epsilon_c$ for a small constant $\epsilon_c > 0$ everywhere except at the source location where $z(\mathbf{r}_c) = 0$. Then the f_1 system is input-to-state stable with respect to the input disturbance x_2. Furthermore, the overall system (3.46) is input-to-state stable with respect to the field input disturbance δ.*

Proof. We already proved in **Theorem** 3.4.1 that the f_2 system is input-to-state stable w.r.t. δ. What remains is to prove that the f_1 system is input-to-state stable w.r.t. x_2. Then we use **Theorem 3** in [52] to conclude that the overall interconnected system is input-to-state stable. For the unforced system, $f_1(x_1, 0)$, $x_2 = \theta_N = 0$. This implies that $\mathbf{n}_l = -\mathbf{N}$, and hence $\langle \mathbf{n}_i, (\mathbf{I} - \mathbf{n}_l \mathbf{n}_l^T)\mathbf{N} \rangle = \langle \mathbf{n}_i, (\mathbf{I} - \mathbf{N}\mathbf{N}^T)\mathbf{N} \rangle = 0$. Therefore

$$\dot{x}_1 = f_1(x_1, 0) = k_f \left[\sum_{i\in\tilde{\mathcal{V}}^f} \left(\langle \hat{\mathbf{u}}_i, \mathbf{n}_i \rangle^2 - ||\hat{\mathbf{u}}_i||^2 \right) + \sum_{(i,l)\in\mathcal{E}^{fl}} \left(\langle \mathbf{n}_i, \mathbf{u}_i \rangle^2 - ||\mathbf{u}_i||^2 \right) \right].$$

Let $V(x_1) = \frac{1}{2} x_1^2$, be a Lyapunov candidate function. Note that V is positive definite and

$V = 0$ if and only if $\theta_f = \theta_l = 0$, which implies that $\mathbf{n}_i = \mathbf{n}_l \; \forall \; i \in \mathcal{V}^f$. Therefore

$$\dot{V} = -k_f x_1 \sum_{i \in \tilde{\mathcal{V}}^f} \left(||\hat{\mathbf{u}}_i||^2 - \langle \hat{\mathbf{u}}_i, \mathbf{n}_i \rangle^2 \right) - k_f x_1 \sum_{(i,l) \in \mathcal{E}^{fl}} (||\mathbf{u}_i||^2 - \langle \mathbf{u}_i, \mathbf{n}_i \rangle^2) \leq 0. \tag{3.51}$$

Recall that $x_1 \geq 0$ and $x_1 = 0$ if and only if $\mathbf{n}_i = \mathbf{n}_j = \mathbf{n}_l$ for all $i, j \in \mathcal{V}^f$. Furthermore, since the undesired equilibrium sets in (3.36) are proved to be unstable in [21], then by LaSalle's Invariance Principal, the unforced system, $\dot{x}_1 = f_1(x_1, 0)$, is asymptotically stable. For the forced system, $\dot{x}_1 = f_1(x_1, x_2)$, we first prove the following lemma

Lemma 3.4.4.

$$\left| \langle \mathbf{n}_i, (\mathbf{I} - \mathbf{n}_l \mathbf{n}_l^T) \mathbf{N} \rangle \right| \leq (\theta_N (2 - \theta_N))^{\frac{1}{2}}. \tag{3.52}$$

Proof. Using the Cauchy-Schwartz inequality

$$\begin{aligned} \left| \langle \mathbf{n}_i, (\mathbf{I} - \mathbf{n}_l \mathbf{n}_l^T) \mathbf{N} \rangle \right| &\leq \left\| (\mathbf{I} - \mathbf{n}_l \mathbf{n}_l^T) \mathbf{N} \right\| \\ &= \langle (\mathbf{I} - \mathbf{n}_l \mathbf{n}_l^T) \mathbf{N}, (\mathbf{I} - \mathbf{n}_l \mathbf{n}_l^T) \mathbf{N} \rangle^{\frac{1}{2}} \\ &= (1 + \langle \mathbf{N}, \mathbf{n}_l \rangle^2 - 2 \langle \mathbf{N}, \mathbf{n}_l \rangle^2)^{\frac{1}{2}} \\ &= (1 - \langle \mathbf{N}, \mathbf{n}_l \rangle^2)^{\frac{1}{2}} \\ &= (\theta_N (2 - \theta_N))^{\frac{1}{2}}. \end{aligned} \tag{3.53}$$

□

Hence, for the forced system, $\dot{x}_1 = f_1(x_1, x_2)$,

$$\begin{aligned}
\dot{V} &\leq x_1\Big[- k_f \sum_{i\in\tilde{\mathcal{V}}^f} \big(||\hat{\mathbf{u}}_i||^2 - \langle \hat{\mathbf{u}}_i, \mathbf{n}_i\rangle^2\big) \\
&- k_f \sum_{(i,l)\in\mathcal{E}^{fl}} (||\mathbf{u}_i||^2 - \langle \mathbf{u}_i, \mathbf{n}_i\rangle^2) + k_l||\nabla z(\mathbf{r}_c)|| \sum_{(i,l)\in\mathcal{E}^{fl}} (\theta_N(2-\theta_N))^{\frac{1}{2}}\Big] \\
&\leq -k_z(1-\epsilon)x_1\Big[\sum_{i\in\tilde{\mathcal{V}}^f} \big(||\hat{\mathbf{u}}_i||^2 - \langle \hat{\mathbf{u}}_i, \mathbf{n}_i\rangle^2\big) + \sum_{(i,l)\in\mathcal{E}^{fl}} (||\mathbf{u}_i||^2 - \langle \mathbf{u}_i, \mathbf{n}_i\rangle^2)\Big], \\
\forall\, &- k_f\epsilon\Big[\sum_{i\in\tilde{\mathcal{V}}^f} \big(||\hat{\mathbf{u}}_i||^2 - \langle \hat{\mathbf{u}}_i, \mathbf{n}_i\rangle^2\big) + \sum_{(i,l)\in\mathcal{E}^{fl}} (||\mathbf{u}_i||^2 - \langle \mathbf{u}_i, \mathbf{n}_i\rangle^2)\Big] \\
&+ k_l||\nabla z(\mathbf{r}_c)|| \sum_{(i,l)\in\mathcal{E}^{fl}} (\theta_N(2-\theta_N))^{\frac{1}{2}} \leq 0
\end{aligned} \tag{3.54}$$

What remains is to find a sufficient function $\rho(|x_2|)$ such that the above condition is satisfied whenever $|x_1| \geq \rho(|x_2|)$. Note that although $x_1, x_2 \geq 0$ by design, we use the absolute function to agree with the standard ISS analysis. Let $\alpha_1(|x_1|) = \alpha_2(|x_1|) = \frac{1}{2}x_1^2$ be class $\mathcal{K}_\infty$ functions. We then obtain

$$\begin{aligned}
&\alpha_1(|x_1|) \leq V(x_1) \leq \alpha_2(|x_1|), \\
&\dot{V} \leq -W(x_1), \quad \forall \quad |x_1| \geq \rho(|x_2|),
\end{aligned} \tag{3.55}$$

where $W(x_1) = W(|x_1|)$ is a class $\mathcal{K}$ function defined by

$$\begin{aligned}
W(|x_1|) = k_z(1-\epsilon)|x_1|\Big[&\sum_{i\in\mathcal{V}^f, l\notin\mathcal{N}_i} \big(||\hat{\mathbf{u}}_i||^2 - \langle \hat{\mathbf{u}}_i, \mathbf{n}_i\rangle^2\big) \\
&+ \sum_{(i,l)\in\mathcal{E}^{fl}} (||\mathbf{u}_i||^2 - \langle \mathbf{u}_i, \mathbf{n}_i\rangle^2)\Big],
\end{aligned} \tag{3.56}$$

and $\rho(|x_2|)$ is a class $\mathcal{K}$ function defined by

$$\rho(|x_2|) = \frac{k_l||\nabla z(\mathbf{r}_c)||}{k_f\epsilon}|\mathcal{E}^{fl}|\sqrt{|x_2|(2-|x_2|)}. \tag{3.57}$$

Therefore, according to *Theorem 4.19* in [68], the forced system $h(t, \theta, \delta)$ is input-to-state stable. Note that, since $x_2 \to 0$ as $\mathbf{n}_l \to -\mathbf{N}$, $\rho(|x_2|) \to 0$. This implies that f_1 system is asymptotically stable. Additionally, since the f_2 system is proved in **Theorem** 3.4.1 to be input-to-state stable with respect to δ, then according to **Theorem 3** in [52], the overall system is input-to-state stable. □

This implies that the SUSD directions of agents $\{4, \cdots, M\}$ asymptotically converge to that of agents $\{1, 2, 3\}$, which in turn converge to the negative direction of the gradient. Since the source is located at the minimum of the field, then **Theorem** 3.4.2 implies that all agents converge to the source location.

3.5 Simulation Results

The SUSD source seeking strategy is simulated for two swarms of 6 and 20 agents as shown in Fig. 3.4. The field is represented by $z(\mathbf{r}_i) = 0.5 * (x_i^2 + y_i^2 + z_i^2)$, which is minimum at the origin, as indicated by a star. The circular red discs represent agents in the set $\{1, 2, 3\}$, while the circular blue discs represent agents in the set $\{4, \cdots, M\}$. The colors of the discs change from dark to light mapping the intensity of the field. The arrows attached to each agent represent the SUSD direction of each agent, while the magenta dotted lines represent the edges of the graph. We use $k_1 = 1.1$ for $i \in \{1, 2, 3\}$ and $k_1 = 1$ for $i \in \{4, \cdots, M\}$, while $k_2 = 0$ for all agents. The separation distances in (3.4) and (3.5) are selected to be $a_{31}^0 = -a_{23}^0 = 0.25$ and $a_{31}^0 = -a_{32}^0 = \frac{\sqrt{2}}{4}$. The separation distance for $w_{i,j}$ in (3.10) is selected to be $\sqrt{0.7}$. The consensus gain of (3.7) is chosen to be $k_f = 8$, which we made it large to balance with the high SUSD speeds especially when the swarm is away from the source. As shown by the swarm trajectories in Fig. 3.4, given the initial random SUSD directions, the strategy successfully steers the swarm toward the source in a relatively short time.

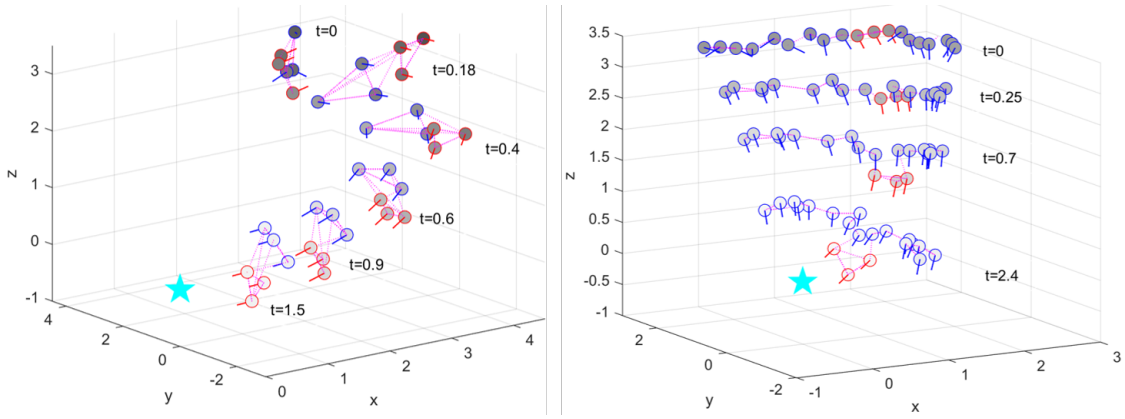


Figure 3.4: Simulations of swarms of 6 and 20 agents. The source location is denoted by the star symbol. The color of each agent represents the intensity of the field at their current locations.

3.6 Conclusion

In this chapter, we presented the SUSD strategy for source seeking in a 3D space. We showed that through a mechanism of synchronizing direction of motion while varying the speed of each agent based only on the measured field value, the strategy successfully steers the swarm toward the minimum of the field.

This chapter contributes to multi-agent systems by providing a method to analyze the collective motion of agents that need to synchronize their direction of motion while modulating their speed. The selection of collective states is a key step to enable convergence and robustness results. Additionally, the resulted Multi-Layer system can be a new framework to describe and analyze the behavior of biological swarms.

CHAPTER 4
INTEGRATING PCA LEARNING ALGORITHM FOR A DISTRIBUTED SOURCE SEEKING

4.1 Introduction

Modeling and control of multi-agent systems is an important problem due to its large variety of potential applications and increasing practical and theoretical challenges. This modeling is often inspired by natural collective behaviors observed for example in schools of fish, flocks of birds, colonies of ants and cultures of bacteria. A remarkable feature these natural behaviors share is their emergence from presumably pure local interactions [1, 2, 3]. This motivates engineers to use swarms of robots in solving complex problems with limited sensing, processing and communicating capabilities [4].

Swarm robotics is a multi-agent system composed of a large number of robots which often have simple capabilities and small sizes. Agents in the swarm interact with each other and with the environment leading to emergent collective behavior. A behavior is considered emergent when it is indirectly and unexpectedly caused by simple local interactions. Agents in a swarm coordinate their motion based on the locally available information. The information is considered to be locally accessible when agents obtain them via measurements of their own sensors or via communication channels with the neighboring agents. Examples of sensor-based local information include inter-agent distances, relative positions, and bearing angles [5].

An important swarm robotics problem is the source seeking where multiple robots are deployed to locate a source of a scalar field. The source is the location where the field has a maximum or minimum value. The field can represent an environmental characteristic including but not limited to, a chemical concentration, a light intensity, or a temperature

value. This problem has various applications ranging from monitoring environmental characteristics to positioning of source signals, and to search and rescue tasks, just to name a few [6, 7].

In this chapter, we propose a Multi-Layer control model composed of an interplay of distributed algorithms for perception and swarming. In the perception layer, each agent learns from the spatial distribution of its neighbors principal directions for motion. These directions are then used in swarming layer to modulate velocities based on the environmental field value. Remarkably, through the interplay of the learning and swarming algorithms, swarms of various sizes and graph structures are able to perform collective source seeking of scalar fields without the need to explicitly estimate the field gradient or explicitly share measurements among the agents.

The different multi-agent control laws developed in the literature to solve this problem generally incorporate a mixture of field gradient and Hessian estimation, extremum seeking control, and weighted consensus laws. Most of the aforementioned control strategies either rely on sharing measurements via communication channels, require maintaining specific spatial formations, or apply only to certain sizes and structures of the interacting graphs. The dependence on the exchange of data through a communication channel is a hard requirement that might be undesired especially in applications with severe limitations such as underwater robotics.

Considering related work to source seeking, in [26], the field gradient is assumed to be known and then a distributed strategy is designed to climb the gradient as well as achieving some desired formations. In [27], agents are required to form a circular formation, and then exchange field measurements via a communication channel to estimate and climb the field gradient. Without knowing the global positions of the agents, a source seeking algorithm is developed in [28], however, it incorporates explicit sharing of field measurement to estimate the gradient. A different gradient-based strategy is studied in [29] where agents autonomously split into multiple subgroups and then each subgroup steers toward a source.

Alternatively, extremum-based source seeking techniques are developed for a single vehicle in a 2D space in [33], and in a 3D space in [34]. The concept is to set the forward velocity as a constant while tuning angular velocities based on an extremum seeking control [32]. Although the approach is simple to implement, the vehicle needs a relatively long distance to travel until the gradient estimation improves. A multi-agent extremum-based source seeking is developed in [35] and [36], however, the agents need to share some estimated parameters. In [37], a strategy is developed for a large number of robots with a complete graph based on a weighted consensus and a Gaussian perturbation. Although it is independent of communication, the agents keep moving randomly in all directions leading to slow, possibly impractical, drift toward the source.

Inspired by a school of fish seeking for darker areas [16], the Speeding-Up, and Slowing-Down (SUSD) strategy is developed for source seeking without gradient estimation in 2D in [30], and in 3D in [31]. However, for networks of more than 2 agents in 2D, and to networks of more than 3 agents in 3D, the agents need to exchange frame components via communication. Differently, these frame components are locally obtained in [9] by incorporating a leader-follower consensus-on-a sphere technique where agents are assumed to be able to measure the velocity directions of their neighbors.

The novel concept in this chapter is in integrating a PCA learning algorithm [13, 14, 15] in the perception layer of the Multi-Layer model through which each agent locally obtains a body frame. This time-varying body frame is then used in the swarming layer by each agent to modulate its motion based only on its instantaneous measurement of the field. The swarming control law is general in that by simple variations of the controller gains, the swarm exhibits different behaviors of source seeking.

The successful elimination of the challenging requirement of explicit estimation and communication is attributed to the locally computed PCA-based body frame and the design of the control law. In particular, the PCA learning algorithm captures changes on the spatial shape and orientation of the swarm which represents an indirect feedback signal

of how the field is affecting the motions of other agents. Additionally, the PCA learning algorithm is local since agents only require knowledge of relative positions. complicating the convergence analysis, the PCA learning algorithm runs on a time scale that is different and faster than that of the swarming algorithm. To overcome this difficulty of multi-scale evolving time, we exploit a singular perturbation framework where the dynamics of the PCA learning and swarming control are viewed as rapidly decaying and slowly varying dynamics, respectively.

This chapter includes four main contributions [10, 12]. The first one is the Multi-Layer model used to design a novel solution to the source seeking problem. The second one is proving the boundedness of the spatial variances of the swarm under complete graphs which renders implicit connectivity-maintenance. The third one is obtaining input-to-state stability results reflecting robust convergence to the source location under complete and incomplete graphs. The last one is validating the proposed model for various source seeking behaviors through simulations and experiments. The experiments are conducted using the Georgia Teach Robotarium [22] and the Georgia Teach Miniature Blimps [23].

The proposed Multi-Layer control model offers a new method that enables robots with limited resources to perform various swarming activities with only local information. In particular, we show that the field measurements of the neighbors are implicitly communicated through the movement of agents. This resembles a communication through behavior which might be useful in modeling information propagation in biological and robotic swarms [47, 46]. Additionally, the use of PCA leverages the value of relative positions in that it extracts more geometrical information about the swarm that might be beneficial for different applications [48, 69].

The rest of this chapter is organized as follows. The problem is formulated in Section 4.2. Then the PCA-based body frame, the design of the control law and the strategy algorithm are presented in Section 4.3. In Section 4.4, we derive the implicit dynamics of the body frame and show how the field measurements are indirectly propagated. Then, stability

analysis and simulation and experimental results are given in Section 4.5 and Section 4.6 respectively. Finally concluding remarks and suggestions of future work are provided in Section 4.7.

4.2 Problem Formulation

4.2.1 Preliminaries

Consider a swarm of M agents described by an undirected visibility graph, $\mathcal{G} \subseteq \mathcal{V} \times \mathcal{E}$ where $\mathcal{V}$ is the set of all agents, and $\mathcal{E}$ is the set of all edges. An undirected edge $(i, j) \in \mathcal{E}$ exists if both agents can sense the relative positions of each other. A graph is connected if for each $i, j \in \mathcal{V}$, there exists a sequence of edges connecting the i^{th} and j^{th} agents. If each agent shares an edge with all other agents, then the graph is complete, otherwise, it is incomplete. The neighbor set of i is defined by $\mathcal{N}_i = \{j | (i, j) \in \mathcal{E}\}$. Additionally, if for each agent $\mathcal{N}_i$ is fixed, the graph is static, otherwise, it is dynamic. We consider the following assumption about the graph.

Assumption 4.2.1. *$\mathcal{G}$ is static, undirected and connected.*

This assumption is to simplify the convergence analysis. However, the design in Section 4.3 is applicable to a broader class of graphs which will be supported by simulation results. Additionally, we will show in Section 4.4 that connectivity is implicitly guaranteed when the graph is complete.

Let $\mathbf{r}_i \in \mathbb{R}^2$ be the position of the i^{th} agent in a 2D space. We require

Assumption 4.2.2. *each i^{th} agent to know the relative positions $\mathbf{r}_j - \mathbf{r}_i$ of all its neighbors, $j \in \mathcal{N}_i$.*

In practice, robots can be equipped with sensors to measure the relative positions of their neighbors with respect to their local frame, which is less challenging than requiring the global positions [5].

Furthermore, suppose each agent measures a scalar field value $z(\mathbf{r}_i) \in \mathbb{R}$ at its current position $\mathbf{r}_i(t)$, where

Assumption 4.2.3. *the field is assumed to be smooth, time invariant and bounded, i.e.* $0 \leq z_{min} \leq z(\mathbf{r}_i) \leq z_{max}$*, and has a unique minimum at the source location* $\mathbf{r}_0$ *where* $z(\mathbf{r}_0) = z_{min}$.

This assumption does not limit the proposed strategy as non-smooth fields may be transformed into smooth fields using for example Stochastic models as in [31].

Let the velocity of each agent be described by

$$\dot{\mathbf{r}}_i = \frac{d\mathbf{r}_i}{dt} = \mathbf{u}_i, \quad i = 1, \cdots, M, \tag{4.1}$$

where $\mathbf{u}_i = \mathbf{u}_i(z_i(t), \{\mathbf{r}_j(t) - \mathbf{r}_i(t)\}_{j \in \mathcal{N}_i})$ is a control law to be designed such that it depends only on the instantaneous field measurement $z_i(t)$, and the relative positions of the neighboring agents $\{\mathbf{r}_j(t) - \mathbf{r}_i(t)\}_{j \in \mathcal{N}_i}$.

4.2.2 Problem Statement

The problem to solve is to design the local control law $\mathbf{u}_i$, such that the swarm autonomously steers toward the source location $\mathbf{r}_0$. A challenging requirement we consider is to solve this problem without explicitly estimating the field gradient and without explicitly communicating field measurements among the agents.

4.3 The Multi-agent System Design

In this section, we first design a locally computed time-varying body frame. We then design a distributed control law to achieve both source seeking and level curve tracking. Finally, we present the Multi-Layer model and summarize the distributed algorithm.

4.3.1 The PCA Body Frame

Principal Component Analysis (PCA) is a statistical method that computes directions of maximum (minimum) variation of a data set [54]. Given a covariance matrix of a data set, its eigenvector corresponding to the maximum (minimum) eigenvalue represents the direction of the maximum (minimum) variance of the data, with the eigenvalue giving the variance of the data along that direction.

For each agent, consider the set of positions defined by $\mathcal{H}_i = \mathcal{N}_i \bigcup \{i\}$. Then the position covariance matrix $\mathbf{C}_i(t) \in \mathbb{R}^{2\times 2}$ observed by each agent is constructed locally as

$$\mathbf{C}_i(t) = \sum_{k\in\mathcal{H}_i} \big(\mathbf{r}_k(t) - \mathbf{r}_{c_i}(t)\big)\big(\mathbf{r}_k(t) - \mathbf{r}_{c_i}(t)\big)^T, \tag{4.2}$$

where $\mathbf{r}_{c_i} = \frac{1}{M_i}\sum_{k\in\mathcal{H}_i} \mathbf{r}_k$ is the center of the swarm as seen by agent i, and $M_i = |\mathcal{H}_i|$. Each agent obtains the principal directions of the spatial distribution of the surrounding agents by computing the eigenvectors of (4.2). We illustrate the concept in Fig. 4.1 wherein agent i has three immediate neighbors. The spatial largest and smallest principal axis are denoted by PC1 and PC2, respectively. Since each agent may have different neighbors, as in the case when the graph is incomplete, then each agent may obtain different principal directions.

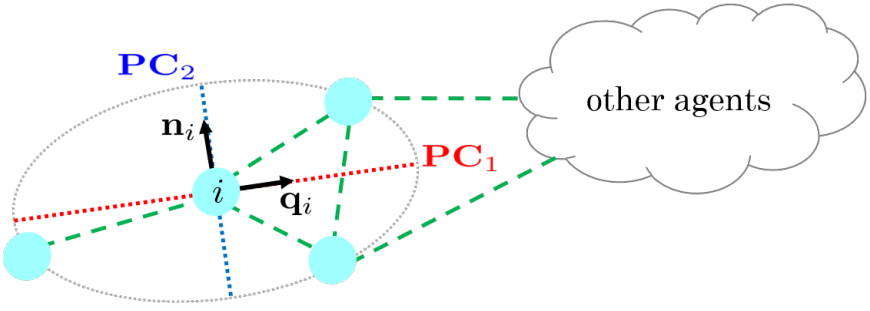


Figure 4.1: The blue dash lines are the edges of the connectivity graph. Agent i sees 3 neighbors from which it forms the largest and smallest principal components PC1 and PC2, respectively.

Different algorithms such as those incorporating singular value decomposition may be

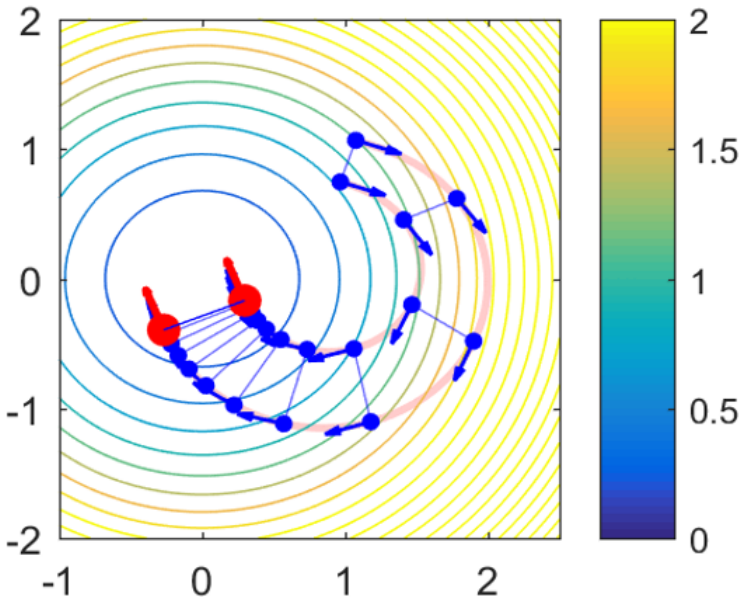


Figure 4.2: The blue arrows are the velocities which turn red at the end time. The circular curves are the level curves of the field which are colored based on the field intensity.

used to compute the eigenvectors of the covariance matrix (4.2). However, to conduct stability analysis, we need a dynamical system that describes the solving process of the principal directions and their evolution over time. For this purpose, we obtain the principal directions by utilizing an unsupervised learning algorithm that is described by a dynamical model called the one-unit Oja PCA Flow model [14, 15].

Let the PCA body-frame of agent i be $(\mathbf{q}_i(t), \mathbf{n}_i(t))$, where $\mathbf{q}_i(t)$ and $\mathbf{n}_i(t)$ are orthonormal vectors in $\mathbb{R}^2$ that represent the principal components of the covariance matrix $\mathbf{C}_i(t)$, corresponding to the largest and smallest eigenvalues, λ_i^q and λ_i^n, respectively. Each agent obtains the body-frame via the Oja PCA Flow model

$$\begin{aligned} \frac{d\mathbf{q}_i(\tau)}{d\tau} &= \Big(\mathbf{I} - \mathbf{q}_i(\tau)\mathbf{q}_i^T(\tau)\Big)\mathbf{C}_i(t)\mathbf{q}_i(\tau) \\ \mathbf{n}_i(\tau) &= \mathbf{R}\mathbf{q}_i(\tau), \end{aligned} \tag{4.3}$$

where $\mathbf{R}$ is a 90° counterclockwise rotation matrix. Observe that we use the argument τ instead of t to emphasize that for any given covariance matrix $\mathbf{C}_i(t)$ at time instant t,

agent i runs (4.3) at a different time scale τ. It is known that the trajectories of (4.3) asymptotically converge to the principal components of $\mathbf{C}_i(t)$ almost everywhere [13, 56]. That is, $\mathbf{q}_i(\tau) \to \mathbf{q}_i(t)$ as $\tau \to \infty$ asymptotically. Importantly, observe that the PCA model (4.3) is scalable to graphs with an arbitrary number of agents and structures. Additionally, the model does not require assigning identities to the agents which enhances the scalability of the algorithm.

Remark 2. *As observed in Fig. 4.1, the principal components are bidirectional vectors, i.e.* $\pm\mathbf{q}_i(t), \pm\mathbf{n}_i(t)$. *However, if all agents use a common initial point at time* $t = 0$ *for their PCA flow (4.3), then we obtain directional principal components.*

4.3.2 The Distributed Control Law

Given the body frame $(\mathbf{q}_i(t), \mathbf{n}_i(t))$ obtained by (4.3), we propose control law

$$\mathbf{u}_i(t) = k_1 z_i(t) \mathbf{n}_i(t), \tag{4.4}$$

where $z_i(t)$ is the instantaneous locally measured field value, and $k_1 \in \mathbb{R}$ is a positive gain used to scale the speed of each agent. To intuitively explain the control law (4.4), we simulate it in Fig. 4.2 for a 2-agent system in a scalar field. In this example, $\mathbf{q}_i = \mathbf{q}$ is along the line-of-sight between the two agents, and $\mathbf{n}_i = \mathbf{n}$ is perpendicular to the line-of-sight. Then agent i speeds up or slows down along the direction $\mathbf{n}$ depending on the local field measurement $z_i(t)$. Hence, the two agents move in the same direction, however, with different speeds leading to a translational velocity and a rotational velocity around the center of the two agents. The rotational velocity settles when the two agents have the same field value, i.e. at the same level curve. At this time, $\mathbf{n}$ is in the negative direction of the field gradient and hence the two agents move towards the minimum of the field.

Remark 3. *Extra terms can be added to the control law (4.4) to, for example, maintain prescribed formations or avoid collisions. In the experimental results, we will show examples*

of formation terms and their effects on the performance. However, we don't consider them in the convergence analysis for the sake of simplicity. Additionally, collision avoidance may be guaranteed using lower level controllers such as barrier functions as in [70].

4.3.3 The Multi-Layer Model

The proposed Multi-Layer model is shown in Fig. 4.3. The two layers run on a different time scale. In the upper perception layer, each agent utilizes the PCA flow (4.3) to learn the body frame components from the spatial distribution of its neighbors. Then, in the swarming layer, each agent applies the distributed control law (4.4) to modulate its speed across the directions obtained by the perception layer.

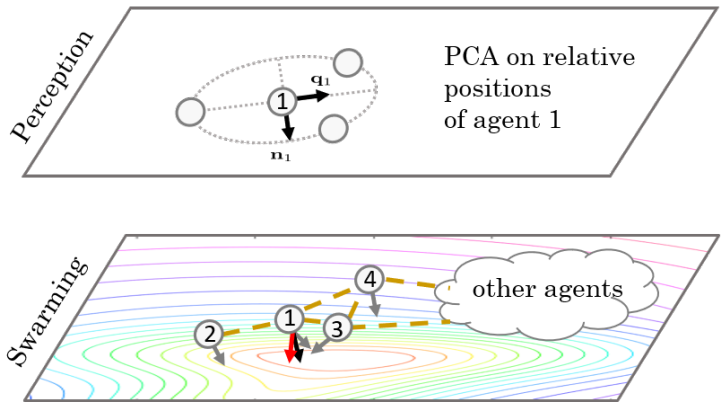


Figure 4.3: The Multi-Layer Model.

A pseudocode description for the source seeking and level curve tracking strategy is given in Algorithm 1. Remarkably, all the steps in the algorithm are locally computed without explicit communication of any values.

4.4 The System Dynamics

The PCA flow system (4.3) describes the dynamics of the PCA components $(\mathbf{n}_i(t), \mathbf{q}_i(t))$ of a given covariance matrix $\mathbf{C}_i(t)$ only in the learning time scale τ. However, to analyze the convergence of the system, we also need the dynamics of the PCA components

Algorithm 1 The Distributed Source Seeking Strategy

```
loop
    for each agent do
        observe neighbor positions
        compute covariance matrix using (4.2)
        compute principal components using (4.3)
        update motion using (4.4)
    end for
end loop
```

$(\mathbf{n}_i(t), \mathbf{q}_i(t))$ in the swarming time scale t. In this section, we derive these implicit dynamics for the general incomplete graph first, and then for the special complete graph. Additionally, at the end of this section, we show that the spatial variances of the swarm are bounded when the graph is complete.

Let $\mathcal{N}_i$ be the neighborhood set of agent i, and $\mathcal{H}_i = \mathcal{N}_i \bigcup \{i\}$, and $M_i = |\mathcal{N}_i| + 1$. The covariance matrix seen by agent i is given by

$$\mathbf{C}_i = \sum_{k \in \mathcal{H}_i} (\mathbf{r}_k - \mathbf{r}_{c,i})(\mathbf{r}_k - \mathbf{r}_{c,i})^T, \tag{4.5}$$

where $\mathbf{r}_{c,i} = \frac{1}{M_i} \sum_{l \in \mathcal{H}_i} \mathbf{r}_l$ is the center of mass as seen by agent i. The covariance matrix satisfies $\mathbf{C}_i \mathbf{n}_i = \lambda_i^n \mathbf{n}_i$, and $\mathbf{C}_i \mathbf{q}_i = \lambda_i^q \mathbf{q}_i$, where λ_i^n and λ_i^q are the smallest and largest eigenvalues of $\mathbf{C}_i$, corresponding to the eigenvectors $\mathbf{n}_i$ and $\mathbf{q}_i$, respectively. Hence, we obtain

$$\dot{\mathbf{C}}_i \mathbf{n}_i + \mathbf{C}_i \dot{\mathbf{n}}_i = \dot{\lambda}_i^n \mathbf{n}_i + \lambda_i^n \dot{\mathbf{n}}_i, \tag{4.6}$$

$$\dot{\mathbf{C}}_i \mathbf{q}_i + \mathbf{C}_i \dot{\mathbf{q}}_i = \dot{\lambda}_i^q \mathbf{q}_i + \lambda_i^q \dot{\mathbf{q}}_i. \tag{4.7}$$

Inner product both sides of (4.6) with the eigenvector $\mathbf{q}_i$, and both sides of (4.7) with the

eigenvector $\mathbf{n}_i$

$$\langle \mathbf{q}_i, \dot{\mathbf{C}}_i \mathbf{n}_i \rangle + \langle \mathbf{q}_i, \mathbf{C}_i \dot{\mathbf{n}}_i \rangle = \dot{\lambda}_i^n \langle \mathbf{q}_i, \mathbf{n}_i \rangle + \lambda_i^n \langle \mathbf{q}_i, \dot{\mathbf{n}}_i \rangle, \tag{4.8}$$

$$\langle \mathbf{n}_i, \dot{\mathbf{C}}_i \mathbf{q}_i \rangle + \langle \mathbf{n}_i, \mathbf{C}_i \dot{\mathbf{q}}_i \rangle = \dot{\lambda}_i^q \langle \mathbf{n}_i, \mathbf{q}_i \rangle + \lambda_i^q \langle \mathbf{n}_i, \dot{\mathbf{q}}_i \rangle. \tag{4.9}$$

Since $\mathbf{C}_i$ is symmetric, then $\dot{\mathbf{C}}_i$ is also symmetric. This implies that

$$\langle \mathbf{q}_i, \mathbf{C}_i \dot{\mathbf{n}}_i \rangle = \langle \mathbf{C}_i \mathbf{q}_i, \dot{\mathbf{n}}_i \rangle = \lambda_i^q \langle \mathbf{q}_i, \dot{\mathbf{n}}_i \rangle \tag{4.10}$$

$$\langle \mathbf{n}_i, \mathbf{C}_i \dot{\mathbf{q}}_i \rangle = \langle \mathbf{C}_i \mathbf{n}_i, \dot{\mathbf{q}}_i \rangle = \lambda_i^n \langle \mathbf{n}_i, \dot{\mathbf{q}}_i \rangle. \tag{4.11}$$

Using (4.10) with the fact that $\langle \mathbf{q}_i, \mathbf{n}_i \rangle = \langle \mathbf{n}_i, \mathbf{q}_i \rangle = 0$, we obtain from (4.8) and (4.9)

$$\langle \mathbf{q}_i, \dot{\mathbf{n}}_i \rangle = -\frac{1}{\lambda_i^q - \lambda_i^n} \langle \mathbf{q}_i, \dot{\mathbf{C}}_i \mathbf{n}_i \rangle, \tag{4.12}$$

$$\langle \mathbf{n}_i, \dot{\mathbf{q}}_i \rangle = \frac{1}{\lambda_i^q - \lambda_i^n} \langle \mathbf{q}_i, \dot{\mathbf{C}}_i \mathbf{n}_i \rangle. \tag{4.13}$$

Since $\mathbf{n}_i$ and $\mathbf{q}_i$ are orthonormal, then we can write

$$\dot{\mathbf{n}}_i = \langle \mathbf{q}_i, \dot{\mathbf{n}}_i \rangle \mathbf{q}_i, \quad \dot{\mathbf{q}}_i = \langle \mathbf{n}_i, \dot{\mathbf{q}}_i \rangle \mathbf{n}_i. \tag{4.14}$$

Substituting (4.12) and (4.13) in (4.14)

$$\dot{\mathbf{n}}_i = -\frac{1}{\lambda_i^q - \lambda_i^n} \langle \mathbf{q}_i, \dot{\mathbf{C}}_i \mathbf{n}_i \rangle \mathbf{q}_i \tag{4.15}$$

$$\dot{\mathbf{q}}_i = \frac{1}{\lambda_i^q - \lambda_i^n} \langle \mathbf{q}_i, \dot{\mathbf{C}}_i \mathbf{n}_i \rangle \mathbf{n}_i \tag{4.16}$$

We then prove the following result for source seeking with general incomplete graphs.

Lemma 4.4.1. *Using the motion dynamics (4.1) along with the PCA flow (4.3) and the control law (4.4), the implicit dynamics of the body frame for source seeking with general incomplete graphs are*

$$\dot{\mathbf{n}}_i = -k_1 \frac{1}{\lambda_i^q - \lambda_i^n} \mathbf{w}_i^T \mathbf{q}_i \mathbf{q}_i - \frac{1}{\lambda_i^q - \lambda_i^n} \sigma_i \mathbf{q}_i, \tag{4.17}$$

$$\dot{\mathbf{q}}_i = k_1 \frac{1}{\lambda_i^q - \lambda_i^n} \mathbf{w}_i^T \mathbf{q}_i \mathbf{n}_i + \frac{1}{\lambda_i^q - \lambda_i^n} \sigma_i \mathbf{n}_i, \tag{4.18}$$

where

$$\mathbf{w}_i = \sum_{k \in \mathcal{H}_i} (z_k \langle \mathbf{n}_k, \mathbf{n}_i \rangle - z_{c,i})(\mathbf{r}_k - \mathbf{r}_{c,i}), \tag{4.19}$$

and

$$\sigma_i = k_1 \sum_{k \in \mathcal{H}_i} \Big(z_k \langle \mathbf{n}_k, \mathbf{q}_i \rangle - z_{c,i} \Big) \langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{n}_i \rangle. \tag{4.20}$$

Proof. Applying time derivative on (4.5)

$$\dot{\mathbf{C}}_i = \sum_{k \in \mathcal{H}_i} [(\dot{\mathbf{r}}_k - \dot{\mathbf{r}}_{c,i})(\mathbf{r}_k - \mathbf{r}_{c,i})^T + (\mathbf{r}_k - \mathbf{r}_{c,i})(\dot{\mathbf{r}}_k - \dot{\mathbf{r}}_{c,i})^T], \tag{4.21}$$

where

$$\mathbf{r}_k - \mathbf{r}_{c,i} = \mathbf{r}_k - \frac{1}{M_i} \sum_{l \in \mathcal{H}_i} \mathbf{r}_l = \frac{1}{M_i} \sum_{j \in \mathcal{N}_i} (\mathbf{r}_k - \mathbf{r}_j). \tag{4.22}$$

Using the control law (4.4), we obtain

$$\begin{aligned}\dot{\mathbf{r}}_k - \dot{\mathbf{r}}_{c,i} &= k_1 z_k \mathbf{n}_k - \frac{k_1}{M_i}\sum_{l\in\mathcal{H}_i} z_l \mathbf{n}_l \\ &= k_1 \frac{M_i - 1}{M_i} z_k \mathbf{n}_k - \frac{k_1}{M_i}\sum_{j\in\mathcal{N}_i} z_j \mathbf{n}_j \\ &= \frac{k_1}{M_i}\sum_{j\in\mathcal{N}_i}(z_k \mathbf{n}_k - z_j \mathbf{n}_j). \end{aligned} \tag{4.23}$$

Substituting (4.23) in (4.21)

$$\dot{\mathbf{C}}_i = \frac{k_1}{M_i}\sum_{k\in\mathcal{H}_i}\sum_{j\in\mathcal{N}_i}\Big[(z_k\mathbf{n}_k - z_j\mathbf{n}_j)(\mathbf{r}_k - \mathbf{r}_{c,i})^T + (\mathbf{r}_k - \mathbf{r}_{c,i})(z_k\mathbf{n}_k - z_j\mathbf{n}_j)^T\Big]. \tag{4.24}$$

Using (4.24), we obtain

$$\begin{aligned}\langle \mathbf{n}_i, \dot{\mathbf{C}}_i\mathbf{q}_i\rangle = &\frac{k_1}{M_i}\sum_{k\in\mathcal{H}_i}\sum_{j\in\mathcal{N}_i}\Big(z_k\langle\mathbf{n}_k,\mathbf{n}_i\rangle - z_j\langle\mathbf{n}_j,\mathbf{n}_i\rangle\Big)\langle\mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{q}_i\rangle \\ &+ \frac{k_1}{M_i}\sum_{k\in\mathcal{H}_i}\sum_{j\in\mathcal{N}_i}\Big(z_k\langle\mathbf{n}_k,\mathbf{q}_i\rangle - z_j\langle\mathbf{n}_j,\mathbf{q}_i\rangle\Big)\langle\mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{n}_i\rangle. \end{aligned} \tag{4.25}$$

But

$$\frac{1}{M_i}\sum_{j\in\mathcal{N}_i}\Big(z_k\langle\mathbf{n}_k,\mathbf{n}_i\rangle - z_j\langle\mathbf{n}_j,\mathbf{n}_i\rangle\Big) = z_k\langle\mathbf{n}_k,\mathbf{n}_i\rangle - \frac{1}{M_i}\sum_{l\in\mathcal{H}_i} z_l\langle\mathbf{n}_l,\mathbf{n}_i\rangle. \tag{4.26}$$

Define $\tilde{z}_{a,i}$ such that

$$\frac{1}{M_i}\sum_{l\in\mathcal{H}_i} z_l\langle\mathbf{n}_l,\mathbf{n}_i\rangle = \tilde{z}_{a,i}\langle\mathbf{n}_k,\mathbf{n}_i\rangle. \tag{4.27}$$

Hence

$$z_k\langle\mathbf{n}_k,\mathbf{n}_i\rangle - \frac{1}{M_i}\sum_{l\in\mathcal{H}_i} z_l\langle\mathbf{n}_l,\mathbf{n}_i\rangle = (z_k - \tilde{z}_{a,i})\langle\mathbf{n}_k,\mathbf{n}_i\rangle. \tag{4.28}$$

Since

$$z_k\langle \mathbf{n}_k, \mathbf{n}_i\rangle - \frac{1}{M_i}\sum_{l\in\mathcal{H}_i} z_l\langle \mathbf{n}_l, \mathbf{n}_i\rangle = \frac{M_i-1}{M_i} z_k\langle \mathbf{n}_k, \mathbf{n}_i\rangle - \frac{1}{M_i}\sum_{j\in\mathcal{N}_i} z_j\langle \mathbf{n}_j, \mathbf{n}_i\rangle$$
$$= \frac{1}{M_i}\sum_{j\in\mathcal{N}_i}(z_k\langle \mathbf{n}_k, \mathbf{n}_i\rangle - z_j\langle \mathbf{n}_j, \mathbf{n}_i\rangle), \tag{4.29}$$

then

$$\frac{1}{M_i}\sum_{j\in\mathcal{N}_i}(z_k\langle \mathbf{n}_k, \mathbf{n}_i\rangle - z_j\langle \mathbf{n}_j, \mathbf{n}_i\rangle) = (z_k - \tilde{z}_{a,i})\langle \mathbf{n}_k, \mathbf{n}_i\rangle. \tag{4.30}$$

Similarly, since

$$z_k\langle \mathbf{n}_k, \mathbf{q}_i\rangle - \frac{1}{M_i}\sum_{l\in\mathcal{H}_i} z_l\langle \mathbf{n}_l, \mathbf{q}_i\rangle = \frac{M_i-1}{M_i} z_k\langle \mathbf{n}_k, \mathbf{q}_i\rangle - \frac{1}{M_i}\sum_{j\in\mathcal{N}_i} z_j\langle \mathbf{n}_j, \mathbf{q}_i\rangle$$
$$= \frac{1}{M_i}\sum_{j\in\mathcal{N}_i}(z_k\langle \mathbf{n}_k, \mathbf{q}_i\rangle - z_j\langle \mathbf{n}_j, \mathbf{q}_i\rangle), \tag{4.31}$$

then

$$\frac{1}{M_i}\sum_{j\in\mathcal{N}_i}(z_k\langle \mathbf{n}_k, \mathbf{q}_i\rangle - z_j\langle \mathbf{n}_j, \mathbf{q}_i\rangle) = (z_k - \tilde{z}_{a,i})\langle \mathbf{n}_k, \mathbf{q}_i\rangle. \tag{4.32}$$

Substituting (4.32) and (4.30) in (4.25) to obtain

$$\langle \mathbf{n}_i, \dot{\mathbf{C}}_i\mathbf{q}_i\rangle = k_1\sum_{k\in\mathcal{H}_i}(z_k - \tilde{z}_{a,i})\Big[\langle \mathbf{n}_k, \mathbf{n}_i\rangle\langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{q}_i\rangle + \langle \mathbf{n}_k, \mathbf{q}_i\rangle\langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{n}_i\rangle\Big]. \tag{4.33}$$

Let: $z_k - \tilde{z}_{a,i} = z_k - z_{c,i} + z_{c,i} - \tilde{z}_{a,i}$. Consequently

$$\sum_{k\in\mathcal{H}_i}(z_k - \tilde{z}_{a,i})\langle \mathbf{n}_k, \mathbf{n}_i\rangle(\mathbf{r}_k - \mathbf{r}_{c,i}) = \sum_{k\in\mathcal{H}_i}(z_k - z_{c,i})\langle \mathbf{n}_k, \mathbf{n}_i\rangle(\mathbf{r}_k - \mathbf{r}_{c,i})$$
$$+ \sum_{k\in\mathcal{H}_i}(z_{c,i} - \tilde{z}_{a,i})\langle \mathbf{n}_k, \mathbf{n}_i\rangle(\mathbf{r}_k - \mathbf{r}_{c,i}). \tag{4.34}$$

But, from (4.27), we have

$$(z_{c,i} - \tilde{z}_{a,i})\langle \mathbf{n}_k, \mathbf{n}_i \rangle = z_{c,i}\langle \mathbf{n}_k, \mathbf{n}_i \rangle - \frac{1}{M_i} \sum_{l \in \mathcal{H}_i} z_l \langle \mathbf{n}_l, \mathbf{n}_i \rangle. \tag{4.35}$$

Then, using (4.35)

$$\begin{aligned} \sum_{k \in \mathcal{H}_i} (z_{c,i} - \tilde{z}_{a,i})\langle \mathbf{n}_k, \mathbf{n}_i \rangle (\mathbf{r}_k - \mathbf{r}_{c,i}) &= \sum_{k \in \mathcal{H}_i} z_{c,i}\langle \mathbf{n}_k, \mathbf{n}_i \rangle (\mathbf{r}_k - \mathbf{r}_{c,i}) \\ &\quad - \frac{1}{M_i} \sum_{k \in \mathcal{H}_i} \sum_{l \in \mathcal{H}_i} z_l \langle \mathbf{n}_l, \mathbf{n}_i \rangle (\mathbf{r}_k - \mathbf{r}_{c,i}) \\ &= \sum_{k \in \mathcal{H}_i} z_{c,i}\langle \mathbf{n}_k, \mathbf{n}_i \rangle (\mathbf{r}_k - \mathbf{r}_{c,i}), \end{aligned} \tag{4.36}$$

where we used the fact that

$$\begin{aligned} \sum_{k \in \mathcal{H}_i} \sum_{l \in \mathcal{H}_i} z_l \langle \mathbf{n}_l, \mathbf{n}_i \rangle (\mathbf{r}_k - \mathbf{r}_{c,i}) &= \left(\sum_{l \in \mathcal{H}_i} z_l \langle \mathbf{n}_l, \mathbf{n}_i \rangle \right) \sum_{k \in \mathcal{H}_i} (\mathbf{r}_k - \mathbf{r}_{c,i}) \\ &= \left(\sum_{l \in \mathcal{H}_i} z_l \langle \mathbf{n}_l, \mathbf{n}_i \rangle \right) \Big(M_i \mathbf{r}_{c,i} - M_i \mathbf{r}_{c,i} \Big) \\ &= 0. \end{aligned} \tag{4.37}$$

Substituting (4.36) in (4.34)

$$\sum_{k \in \mathcal{H}_i} (z_k - \tilde{z}_{a,i})\langle \mathbf{n}_k, \mathbf{n}_i \rangle (\mathbf{r}_k - \mathbf{r}_{c,i}) = \sum_{k \in \mathcal{H}_i} z_k \langle \mathbf{n}_k, \mathbf{n}_i \rangle (\mathbf{r}_k - \mathbf{r}_{c,i}) \tag{4.38}$$

We then add $0 = \sum_{k \in \mathcal{H}_i} z_c(\mathbf{r}_k - \mathbf{r}_{c,i})$ to (4.38) to obtain

$$\sum_{k \in \mathcal{H}_i} (z_k - \tilde{z}_{a,i})\langle \mathbf{n}_k, \mathbf{n}_i \rangle (\mathbf{r}_k - \mathbf{r}_{c,i}) = \sum_{k \in \mathcal{H}_i} (z_k \langle \mathbf{n}_k, \mathbf{n}_i \rangle - z_{c,i})(\mathbf{r}_k - \mathbf{r}_{c,i}) \tag{4.39}$$

Similarly,

$$\sum_{k\in\mathcal{H}_i}(z_k-\tilde{z}_{a,i})\langle\mathbf{n}_k,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i}) = \sum_{k\in\mathcal{H}_i}(z_k-z_{c,i})\langle\mathbf{n}_k,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i}) + \sum_{k\in\mathcal{H}_i}(z_{c,i}-\tilde{z}_{a,i})\langle\mathbf{n}_k,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i}). \tag{4.40}$$

But, from (4.27), we have

$$(z_{c,i}-\tilde{z}_{a,i})\langle\mathbf{n}_k,\mathbf{q}_i\rangle = z_{c,i}\langle\mathbf{n}_k,\mathbf{q}_i\rangle - \frac{1}{M_i}\sum_{l\in\mathcal{H}_i} z_l\langle\mathbf{n}_l,\mathbf{q}_i\rangle. \tag{4.41}$$

Then, using (4.41)

$$\begin{aligned}\sum_{k\in\mathcal{H}_i}(z_{c,i}-\tilde{z}_{a,i})\langle\mathbf{n}_k,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i}) &= \sum_{k\in\mathcal{H}_i} z_{c,i}\langle\mathbf{n}_k,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i}) \\ &\quad - \frac{1}{M_i}\sum_{k\in\mathcal{H}_i}\sum_{l\in\mathcal{H}_i} z_l\langle\mathbf{n}_l,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i}) \\ &= \sum_{k\in\mathcal{H}_i} z_{c,i}\langle\mathbf{n}_k,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i}),\end{aligned} \tag{4.42}$$

where we used the fact that

$$\begin{aligned}\sum_{k\in\mathcal{H}_i}\sum_{l\in\mathcal{H}_i} z_l\langle\mathbf{n}_l,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i}) &= \left(\sum_{l\in\mathcal{H}_i} z_l\langle\mathbf{n}_l,\mathbf{q}_i\rangle\right)\sum_{k\in\mathcal{H}_i}(\mathbf{r}_k-\mathbf{r}_{c,i}) \\ &= \left(\sum_{l\in\mathcal{H}_i} z_l\langle\mathbf{n}_l,\mathbf{q}_i\rangle\right)\Big(M_i\mathbf{r}_{c,i}-M_i\mathbf{r}_{c,i}\Big) \\ &= 0.\end{aligned} \tag{4.43}$$

Substituting (4.42) in (4.40)

$$\sum_{k\in\mathcal{H}_i}(z_k-\tilde{z}_{a,i})\langle\mathbf{n}_k,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i}) = \sum_{k\in\mathcal{H}_i} z_k\langle\mathbf{n}_k,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i}) \tag{4.44}$$

We then add $0 = \sum_{k\in\mathcal{H}_i} z_c(\mathbf{r}_k - \mathbf{r}_{c,i})$ to (4.44) to obtain

$$\sum_{k\in\mathcal{H}_i}(z_k - \tilde{z}_{a,i})\langle \mathbf{n}_k, \mathbf{q}_i\rangle(\mathbf{r}_k - \mathbf{r}_{c,i}) = \sum_{k\in\mathcal{H}_i}(z_k\langle \mathbf{n}_k, \mathbf{q}_i\rangle - z_{c,i})(\mathbf{r}_k - \mathbf{r}_{c,i}) \tag{4.45}$$

Substituting (4.39) and (4.42) in (4.33)

$$\begin{aligned}\langle \mathbf{n}_i, \dot{\mathbf{C}}_i\mathbf{q}_i\rangle = k_1 \sum_{k\in\mathcal{H}_i}\Big(z_k\langle \mathbf{n}_k, \mathbf{n}_i\rangle - z_{c,i}\Big)\langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{q}_i\rangle \\ + k_1 \sum_{k\in\mathcal{H}_i}\Big(z_k\langle \mathbf{n}_k, \mathbf{q}_i\rangle - z_{c,i}\Big)\langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{n}_i\rangle.\end{aligned} \tag{4.46}$$

Define

$$\mathbf{w}_i = \sum_{k\in\mathcal{H}_i}\Big(z_k\langle \mathbf{n}_k, \mathbf{n}_i\rangle - z_{c,i}\Big)(\mathbf{r}_k - \mathbf{r}_{c,i}). \tag{4.47}$$

Then, substituting (4.47) in (4.46) and using (4.15) and (4.16), we obtain

$$\dot{\mathbf{n}}_i = -k_1\frac{1}{\lambda_i^q - \lambda_i^n}\mathbf{w}_i^T\mathbf{q}_i\mathbf{q}_i - \frac{1}{\lambda_i^q - \lambda_i^n}\sigma_i\mathbf{q}_i, \tag{4.48}$$

$$\dot{\mathbf{q}}_i = k_1\frac{1}{\lambda_i^q - \lambda_i^n}\mathbf{w}_i^T\mathbf{q}_i\mathbf{n}_i + \frac{1}{\lambda_i^q - \lambda_i^n}\sigma_i\mathbf{n}_i \tag{4.49}$$

where

$$\sigma_i = k_1 \sum_{k\in\mathcal{H}_i}\Big(z_k\langle \mathbf{n}_k, \mathbf{q}_i\rangle - z_{c,i}\Big)\langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{n}_i\rangle. \tag{4.50}$$

□

Note that $\sigma_i \to \mathbf{0}$ as $\langle \mathbf{n}_i, \mathbf{n}_j\rangle \to 1\ \forall j \in \mathcal{N}_i$.

4.4.1 Source Seeking with Incomplete Graphs

Since the field is smooth, then using Taylor expansion, we express

$$z_k - z_{c,i} = \langle \mathbf{r}_k - \mathbf{r}_{c,i}, \nabla z_i \rangle + \nu_k, \tag{4.51}$$

where $\nabla z_i = \nabla z(\mathbf{r}_{c,i})$ is the local gradient in the vicinity of the center $\mathbf{r}_{c,i}$ observed by agent i, and ν_k represents the higher order terms.

When the graph is incomplete, since $\nabla z_i = \nabla z(\mathbf{r}_{c,i})$ is gradient at the local center, then we may ignore the higher order term, ν_k, in (4.51), especially if the neighboring agents are close enough to each other. In this case, we have the following result.

Corollary 4.4.1. *Suppose the field is linear at the vicinity of the local center, $\mathbf{r}_{c,i}$. Using the motion dynamics (4.1) along with the PCA flow (4.3) and the control law (4.4), the implicit dynamics of the body frame for source seeking with incomplete graphs are*

$$\dot{\mathbf{n}}_i = -k_1 ||\nabla z_i|| \frac{\lambda_i^q}{\lambda_i^q - \lambda_i^n} (\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T) \mathbf{N}_i + \mathcal{E}_i \mathbf{q}_i, \tag{4.52}$$

$$\dot{\mathbf{q}}_i = k_1 ||\nabla z_i|| \frac{\lambda_i^q}{\lambda_i^q - \lambda_i^n} \langle \mathbf{N}_i, \mathbf{q}_i \rangle \mathbf{n}_i + \mathcal{E}_i \mathbf{n}_i, \tag{4.53}$$

where $\mathbf{N}_i = \frac{\nabla z_i}{||\nabla z_i||}$, *and* $\mathcal{E}_i = k_1 \frac{1}{\lambda_i^q - \lambda_i^n} [e_i + \sum_{k \in \mathcal{H}_i} z_{c,i} \langle \mathbf{n}_k, \mathbf{n}_i \rangle \langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{q}_i \rangle - \sum_{k \in \mathcal{H}_i} z_{k,i} \langle \mathbf{n}_k, \mathbf{q}_i \rangle \langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{n}_i \rangle]$.

Proof. Multiplying both sides of (4.51) by $\langle \mathbf{n}_k, \mathbf{n}_i \rangle$ yields

$$z_k \langle \mathbf{n}_k, \mathbf{n}_i \rangle = z_{c,i} \langle \mathbf{n}_k, \mathbf{n}_i \rangle + \langle \mathbf{r}_k - \mathbf{r}_{c,i}, \nabla z_i \rangle \langle \mathbf{n}_k, \mathbf{n}_i \rangle. \tag{4.54}$$

Using (4.54) and the fact that $\sum_{k\in\mathcal{H}_i} z_{c,i}(\mathbf{r}_k - \mathbf{r}_{c,i}) = 0$, we obtain

$$\begin{aligned}
&\sum_{k\in\mathcal{H}_i} (z_k\langle \mathbf{n}_k, \mathbf{n}_i\rangle - z_{c,i})\langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{q}_i\rangle \mathbf{q}_i \\
&= \sum_{k\in\mathcal{H}_i} \langle \mathbf{n}_k, \mathbf{n}_i\rangle \Big[\langle \mathbf{r}_k - \mathbf{r}_{c,i}, \nabla z_i\rangle - z_{c,i}\Big] \langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{q}_i\rangle \mathbf{q}_i \\
&= \langle \nabla z_i, \hat{\mathbf{C}}_i \mathbf{q}_i\rangle \mathbf{q}_i - \sum_{k\in\mathcal{H}_i} z_{c,i}\langle \mathbf{n}_k, \mathbf{n}_i\rangle \langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{q}_i\rangle \mathbf{q}_i, \qquad (4.55)
\end{aligned}$$

where $\hat{\mathbf{C}}_i = \sum_{k\in\mathcal{H}_i} \langle \mathbf{n}_k, \mathbf{n}_i\rangle (\mathbf{r}_k - \mathbf{r}_{c,i})(\mathbf{r}_k - \mathbf{r}_{c,i})^T$ is a weighted covariance matrix in which each agent has a mass given by $\langle \mathbf{n}_k, \mathbf{n}_i\rangle$. Let $\langle \nabla z_i, \hat{\mathbf{C}}_i \mathbf{q}_i\rangle = \langle \nabla z_i, \mathbf{C}_i \mathbf{q}_i\rangle + e_i = \lambda_i^q \langle \nabla z_i, \mathbf{q}_i\rangle + e_i$, where the error $e_i \to 0$ as $\langle \mathbf{n}_k, \mathbf{n}_i\rangle \to 1$ for all $k \in \mathcal{H}_i$. Then, substituting (4.55) in (4.47) and (4.20), along with (4.48) and (4.49), we obtain the claimed (4.52) and (4.53). □

Note that $\mathcal{E}_i \to 0$ as $\langle \mathbf{n}_k, \mathbf{n}_i\rangle \to 1$ for all $k \in \mathcal{H}_i$. Additionally, for all $\langle \mathbf{n}_k, \mathbf{n}_i\rangle \in [a, 1]$ where $a > 0$, $\mathcal{E}_i$ can be made arbitrary small by making a large enough.

4.4.2 Source Seeking with Complete Graphs

On the other hand, when the graph is complete, then all agents compute the same body frame $(\mathbf{n}_i, \mathbf{q}_i) = (\mathbf{n}, \mathbf{q})$. In this case we obtain the following result.

Corollary 4.4.2. *Using the motion dynamics (4.1) along with the PCA flow (4.3) and the control law (4.4), the implicit dynamics of the body frame for source seeking with incomplete graphs are*

$$\dot{\mathbf{n}} = -\frac{k_1}{\lambda^q - \lambda^n} \mathbf{w}^T \mathbf{q} \mathbf{q} \qquad (4.56)$$

$$\dot{\mathbf{q}} = +\frac{k_1}{\lambda^q - \lambda^n} \mathbf{w}^T \mathbf{q} \mathbf{n}. \qquad (4.57)$$

Proof. When the graph is complete, then each agent computes the same covariance matrix

$$\mathbf{C}_i = \mathbf{C} = \sum_{k=1}^{M} (\mathbf{r}_k - \mathbf{r}_c)(\mathbf{r}_k - \mathbf{r}_c)^T, \tag{4.58}$$

where all the agents see the same center $\mathbf{r}_c = \frac{1}{M}\sum_{k=1}^{M} \mathbf{r}_k$. This implies that $\mathbf{n}_i = \mathbf{n}_j = \mathbf{n}$, and $\mathbf{q}_i = \mathbf{q}_j = \mathbf{n}$ for all i, j. Hence, $\langle \mathbf{n}_i, \mathbf{q}_j \rangle = 0$, and $\langle \mathbf{q}_i, \mathbf{q}_j \rangle = 1$ for all i, j, and thus

$$\sigma_i = -k_1 z_c \sum_{k=1}^{M} (\mathbf{r}_k - \mathbf{r}_c) = -k_1 z_c (M\mathbf{r}_c - M\mathbf{r}_c) = 0. \tag{4.59}$$

Additionally, using (4.47)

$$\mathbf{w} = \sum_{k=1}^{M} (z_k - z_c)(\mathbf{r}_k - \mathbf{r}_c). \tag{4.60}$$

Substituting (4.59) and (4.60) in (4.48) and (4.49) to obtain

$$\dot{\mathbf{n}} = -\frac{k_1}{\lambda^q - \lambda^n} \mathbf{w}^T \mathbf{q}\mathbf{q} \tag{4.61}$$

$$\dot{\mathbf{q}} = +\frac{k_1}{\lambda^q - \lambda^n} \mathbf{w}^T \mathbf{q}\mathbf{n}. \tag{4.62}$$

□

Furthermore, for a complete graph, we may view the entire swarm as a super agent and define $\nabla z_c = \nabla z(\mathbf{r}_c)$ to be the field gradient at the center of the swarm. Then, without ignoring the higher order term, ν_k in (4.51), we obtain the following result.

Lemma 4.4.2. *Using the motion dynamics (4.1) along with the PCA flow (4.3) and the control law (4.4), the implicit dynamics of the body frame for source seeking with complete*

graphs are

$$\dot{\mathbf{n}} = -\frac{k_1}{\lambda^q - \lambda^n}\Big(||\nabla z_c||\lambda^q\langle \mathbf{N}, \mathbf{q}\rangle + \sum_{k=1}^{M} \nu_k \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q}\rangle\Big)\mathbf{q}, \tag{4.63}$$

$$\dot{\mathbf{q}} = \frac{k_1}{\lambda^q - \lambda^n}\Big(||\nabla z_c||\lambda^q\langle \mathbf{N}, \mathbf{q}\rangle + \sum_{k=1}^{M} \nu_k \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q}\rangle\Big)\mathbf{n}, \tag{4.64}$$

where $\mathbf{N} = \frac{\nabla z_c}{||\nabla z_c||}$.

Proof. Substituting (4.51) in (4.60) yields

$$\mathbf{w} = \sum_{k=1}^{M} [\langle \mathbf{r}_k - \mathbf{r}_c, \nabla z_c\rangle(\mathbf{r}_k - \mathbf{r}_c) + \nu_k(\mathbf{r}_k - \mathbf{r}_c)]. \tag{4.65}$$

But

$$\sum_{k=1}^{M} \langle \mathbf{r}_k - \mathbf{r}_c, \nabla \mathbf{z_c}\rangle(\mathbf{r}_k - \mathbf{r}_c) = \sum_{k=1}^{M} (\mathbf{r}_k - \mathbf{r}_c)(\mathbf{r}_k - \mathbf{r}_c)^T \nabla z_c = \mathbf{C}\nabla z_c. \tag{4.66}$$

Hence

$$\mathbf{w} = \mathbf{C}\nabla z_c + \sum_{k=1}^{M} \nu_k(\mathbf{r}_k - \mathbf{r}_c). \tag{4.67}$$

Then, substituting (4.67) in (4.56) yields

$$\begin{aligned}\dot{\mathbf{n}} &= -\frac{k_1}{\lambda^q - \lambda^n}\nabla \mathbf{z_c}^T \mathbf{C}\mathbf{q}\mathbf{q} - \frac{k_1}{\lambda^q - \lambda^n}\sum_{k=1}^{M} \nu_k \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q}\rangle \mathbf{q} \\ &= -\frac{k_1}{\lambda^q - \lambda^n}\Big(||\nabla z_c||\lambda^q\langle \mathbf{N}, \mathbf{q}\rangle + \sum_{k=1}^{M} \nu_k \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q}\rangle\Big)\mathbf{q}, \end{aligned} \tag{4.68}$$

where $\mathbf{N} = \frac{\nabla z_c}{||\nabla z_c||}$ is a unit length vector representing the direction of the field gradient at

the center of the swarm. Similarly, using (4.65) and (4.49), we obtain

$$\dot{\mathbf{q}} = \frac{k_1}{\lambda^q - \lambda^n}\Big(||\nabla z_c||\lambda^q\langle \mathbf{N}, \mathbf{q}\rangle + \sum_{k=1}^{M} \nu_k \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q}\rangle\Big)\mathbf{n}. \tag{4.69}$$

□

Since $\mathbf{n}$ and $\mathbf{q}$ are orthonormal, we can write $\mathbf{q}\mathbf{q}^T = \mathbf{I} - \mathbf{n}\mathbf{n}^T$. Hence, we can reform (4.63) as

$$\begin{aligned}\dot{\mathbf{n}} = &-k_1\frac{\lambda^q}{\lambda^q - \lambda^n}||\nabla z_c||(\mathbf{I} - \mathbf{n}\mathbf{n}^T)\mathbf{N} \\ &- \frac{k_1}{\lambda^q - \lambda^n}\sum_{k=1}^{M} \nu_k \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q}\rangle \mathbf{q}\end{aligned} \tag{4.70}$$

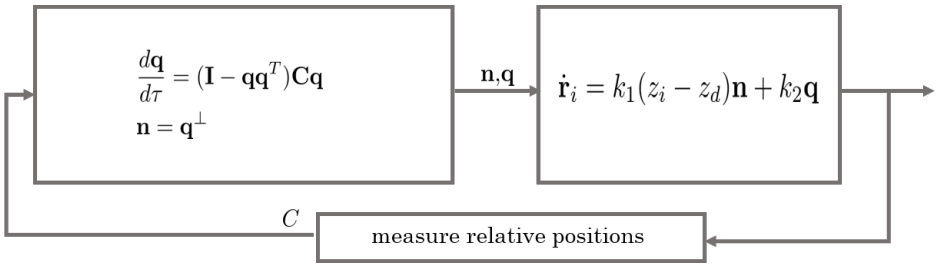
Observe that the term $\sum_{k=1}^{M} \nu_k \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q}\rangle$ in (4.63) and (4.64) vanishes when $\nu_k = \nu$ for all agents. i.e. the field is linear in the vicinity of the swarm's center, or when the agents are at the same level curve.

Remark 4. *The first term in (4.70) represents a consensus-on-a sphere control law [21]. This is remarkable since although we are explicitly applying (4.4) with (4.3), the direction* $\mathbf{n}$ *is implicitly tracking the negative direction of the gradient* $-\mathbf{N}$*. This idea is illustrated in Fig. 4.4 where we draw the following insights.*

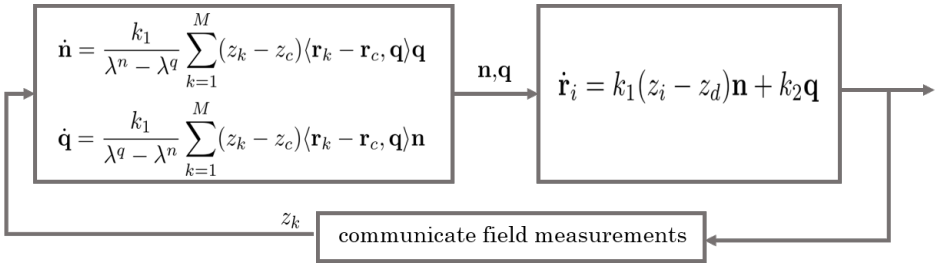
1. *The implicit dynamics in Fig. 4.4 (b) represent a new gradient-free strategy, however, it requires sharing field measurements. Remarkably, the PCA flow totally eliminates the need to explicitly communicate, as demonstrated in Fig. 4.4 (a).*

2. *The presence of* $\sum_i (z_i - z_a)$ *in the implicit dynamics in Fig. 4.4 (b) reveals that the field measurements are indirectly communicated, which elegantly resembles a communication through behavior.*

3. *In view of the implicit dynamics in Fig. 4.4 (b),* $\dot{\mathbf{n}} = \frac{d\mathbf{n}}{dt}$ *is obtained by an integration over the field measurements. This reflects the robustness and smoothness of the*



(a) Explicit dynamics.



(b) Implicit dynamics.

Figure 4.4: The explicit and implicit dynamics of the strategy.

strategy against noisy measurements.

4. *The spatial gain* $\frac{1}{\lambda_q - \lambda_n}$ *in the implicit dynamics captures the effect of the shape of the swarm. When agents are more spatially distributed, i.e. encounter more diverse measurements, then the gain is higher and hence the swarm steers faster.*

We conclude this section by the following result

Lemma 4.4.3. *For a complete graph, for both source seeking and level curve tracking,* $\lambda^n(t) \leq \lambda^q(t) = \lambda^q(t_0)$*, where* $\lambda^q(t_0)$ *is the initial maximum variance of the spatial distribution of the agents.*

Proof. By the definition of PCA [54], for all unit vectors $\mathbf{u} \in \mathbf{R}^2$

$$\lambda^n = \underset{\mathbf{u}}{\operatorname{argmin}}\, \mathbf{u}^T\mathbf{C}\mathbf{u} \tag{4.71}$$

is the smallest directional variance of the spatial distribution, and

$$\lambda^q = \underset{\mathbf{u}}{\operatorname{argmax}}\, \mathbf{u}^T \mathbf{C} \mathbf{u} \tag{4.72}$$

is the largest directional variance of the spatial distribution. This implies that $\lambda^n(t) \leq \lambda^q(t)$. What remains is to show that $\dot{\lambda}^q = 0$. Taking the time derivative of

$$\lambda^q = \mathbf{q}^T \mathbf{C} \mathbf{q} = \sum_{i=1}^{M} \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{q} \rangle^2, \tag{4.73}$$

we obtain

$$\dot{\lambda}^q = 2 \sum_{i=1}^{M} \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{q} \rangle (\langle \dot{\mathbf{r}}_i - \dot{\mathbf{r}}_c, \mathbf{q} \rangle + \langle \mathbf{r}_i - \mathbf{r}_c, \dot{\mathbf{q}} \rangle) \tag{4.74}$$

Using (4.23), for a complete graph

$$\dot{\mathbf{r}}_i - \dot{\mathbf{r}}_c = k_1 (z_i - z_a) \mathbf{n}, \tag{4.75}$$

where $z_a = \frac{1}{M} \sum_{i=1}^{M} z_i$ is the average field measurement. Hence

$$\langle \dot{\mathbf{r}}_i - \dot{\mathbf{r}}_c, \mathbf{q} \rangle = k_1 (z_i^d - z_a^d) \langle \mathbf{n}, \mathbf{q} \rangle = 0. \tag{4.76}$$

Therefore

$$\dot{\lambda}^q = 2 \sum_{i=1}^{M} \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{q} \rangle \langle \mathbf{r}_i - \mathbf{r}_c, \dot{\mathbf{q}} \rangle = 2 \langle \mathbf{C}\mathbf{q}, \dot{\mathbf{q}} \rangle = 2 \lambda^q \langle \mathbf{q}, \dot{\mathbf{q}} \rangle = 0, \tag{4.77}$$

where using (4.64), we obtain

$$\langle \mathbf{q}, \dot{\mathbf{q}} \rangle = \frac{k_1}{\lambda^q - \lambda^n} \Big(||\nabla z_c|| \lambda^q \langle \mathbf{N}, \mathbf{q} \rangle + \sum_{k=1}^{M} \nu_k \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q} \rangle \Big) \langle \mathbf{q}, \mathbf{n} \rangle = 0. \tag{4.78}$$

Similarly, taking the time derivative of

$$\lambda^n = \mathbf{n}^T \mathbf{C} \mathbf{n} = \sum_{i=1}^{M} \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n} \rangle^2, \tag{4.79}$$

we obtain

$$\dot{\lambda}^n = 2 \sum_{i=1}^{M} \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n} \rangle (\langle \dot{\mathbf{r}}_i - \dot{\mathbf{r}}_c, \mathbf{n} \rangle + \langle \mathbf{r}_i - \mathbf{r}_c, \dot{\mathbf{n}} \rangle) \tag{4.80}$$

But using (4.75)

$$\langle \dot{\mathbf{r}}_i - \dot{\mathbf{r}}_c, \mathbf{n} \rangle = k_1 (z_i - z_a) \langle \mathbf{n}, \mathbf{n} \rangle = k_1 (z_i - z_a), \tag{4.81}$$

and

$$\sum_{i=1}^{M} \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n} \rangle \langle \mathbf{r}_i - \mathbf{r}_c, \dot{\mathbf{n}} \rangle = 2 \langle \mathbf{C}\mathbf{n}, \dot{\mathbf{n}} \rangle = 2 \lambda^n \langle \mathbf{n}, \dot{\mathbf{n}} \rangle = 0, \tag{4.82}$$

where $\langle \mathbf{n}, \dot{\mathbf{n}} \rangle = 0$ is obtained from (4.56). Substituting (4.81) and (4.81) into (4.80) yields

$$\dot{\lambda}^n = 2 k_1 \sum_{i=1}^{M} (z_i - z_a) \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n} \rangle. \tag{4.83}$$

Additionally, using (4.51) and the fact that $\sum_{k=1}^{M} \langle \mathbf{r}_k - \mathbf{r}_c, \nabla z_c \rangle = 0$, we write

$$z_a = \frac{1}{M} \sum_{k=1}^{M} z_k = z_c + \frac{1}{M} \sum_{k=1}^{M} \langle \mathbf{r}_k - \mathbf{r}_c, \nabla z_c \rangle + \frac{1}{M} \sum_{k=1}^{M} \nu_k = z_c + \frac{1}{M} \sum_{k=1}^{M} \nu_k. \tag{4.84}$$

Hence

$$\dot{\lambda}^n = 2 k_1 \sum_{i=1}^{M} (z_i - z_c) \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n} \rangle, \tag{4.85}$$

where we used the fact that $\sum_{i=1}^{M} \sum_{k=1}^{M} \nu_k \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n} \rangle = \sum_{k=1}^{M} \nu_k \sum_{i=1}^{M} \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n} \rangle = 0$.

Finally, using (4.51) for $(z_i - z_c)$

$$\begin{aligned}\dot{\lambda}^n &= 2k_1||\nabla z_c||\mathbf{N}^T\sum_{i=1}^{M}(\mathbf{r}_i-\mathbf{r}_c)(\mathbf{r}_i-\mathbf{r}_c)^T\mathbf{n} + 2k_1\sum_{i=1}^{M}\nu_i\langle\mathbf{r}_i-\mathbf{r}_c,\mathbf{n}\rangle \\ &= 2k_1\lambda^n||\nabla z_c||\langle\mathbf{N},\mathbf{n}\rangle + 2k_1\sum_{i=1}^{M}\nu_i\langle\mathbf{r}_i-\mathbf{r}_c,\mathbf{n}\rangle. \qquad (4.86)\end{aligned}$$

From (4.86), λ^n will increase or decrease, hence the shape will stretch or shrink along the $\mathbf{n}$ direction, depending on the signs of $\langle\mathbf{N},\mathbf{n}\rangle$ and $\sum_{i=1}^{M}\nu_i\langle\mathbf{r}_i-\mathbf{r}_c,\mathbf{n}\rangle$. However, if it increases, it will do so only up to $\lambda^n = \lambda^q$. At this point, according to the definitions (4.71) and (4.72), the PCA flow will interchange $\mathbf{n}$ and $\mathbf{q}$ and hence the swarm performs a turn of at most 90°. □

As a result from **Lemma 4.4.3**, we can conclude that the connectivity of the graph is maintained implicitly.

4.5 Convergence Analysis

We want to prove that using the PCA flow (4.3), the control (4.4) steers the agents toward the source. Recall that the PCA flow (4.3) runs in the time scale τ, while the swarming control law (4.4) runs in the time scale t. That is, for each time instance t, each agent runs (4.3) for some time τ. To overcome this difficulty, we formulate a singular perturbation problem as follows. Let the relationships between the swarming time t, and the PCA learning time τ be of the form $\frac{dt}{d\tau} = \epsilon$, where $\epsilon \in (0,1)$. This implies that $\tau = \frac{t-t_0}{\epsilon}$, where $\tau_0 = 0$. In other words, τ is stretched as $\epsilon \to 0$, and shrunk as $\epsilon \to 1$. Using this relationship, the PCA learning and swarming dynamics in the singular perturbation framework are

$$\frac{d\mathbf{r}_i}{dt} = k_1 z_i(t)\mathbf{n}_i(t), \quad \forall i, \qquad (4.87)$$

$$\epsilon\frac{d\mathbf{q}_i}{dt} = (\mathbf{I} - \mathbf{q}_i(\tau)\mathbf{q}_i^T(\tau))\mathbf{C}_i(t)\mathbf{q}_i(\tau), \quad \forall i, \qquad (4.88)$$

where the swarming dynamics (4.87) is viewed as a slow system, and the PCA learning dynamics (4.88) is viewed as a fast system. In this framework, we first let $\epsilon = 0$ in (4.88) to obtain decoupled reduced and boundary systems. We then analyze the stability of the origins of these decoupled systems. Finally, we derive $\epsilon^* \in (0, 1)$ such that for all $\epsilon \leq \epsilon^*$, the stability results of the reduced and boundary systems hold for the original slow (4.87) and fast (4.88) systems. Practically, ϵ^* reveals how rapid the fast systems must be in order for the system to converge to the desired equilibrium. In what follows, we first analyze the source seeking and level curve tracking in complete graphs and then generalize the source seeking to incomplete graphs.

4.5.1 Source Seeking with a Complete Graph

As the graph is complete, we may view the swarm as one body where its individuals are moving in the same direction but with different speeds depending on their field measurements. Since in view of (4.87), the dynamics of the center of the swarm $\dot{\mathbf{r}}_c$ does not explicitly depend on $\mathbf{r}_c$, we will show that $\mathbf{r}_c$ converges to the source location $\mathbf{r}_s$ implicitly. In particular, we will show that the field measurement at the center z_c decreases and motion direction $\mathbf{n}$ converges $-\mathbf{N}$, i.e. the negative direction of the field gradient.

The Reduced System

Taking the time derivative of z_c, and using (4.87), we obtain

$$\dot{z}_c = \langle \nabla z_c, \dot{\mathbf{r}}_c \rangle = \langle \nabla z_c, \frac{k_1}{M} \sum_{k=1}^{M} z_k \mathbf{n} \rangle. \tag{4.89}$$

However, using (4.51)

$$\frac{1}{M} \sum_{k=1}^{M} z_k = \frac{1}{M} \sum_{k=1}^{M} \Big[z_c + \langle \nabla z_c, \mathbf{r}_k - \mathbf{r}_c \rangle + \nu_k \Big] = z_c + \frac{1}{M} \sum_{k=1}^{M} \nu_k, \tag{4.90}$$

where we used the fact $\langle \nabla z_c, \sum_{k=1}^{M}(\mathbf{r}_k - \mathbf{r}_c)\rangle = 0$. Hence

$$\dot{z}_c = k_1 ||\nabla z_c|| \langle \mathbf{N}, \mathbf{n}\rangle \Big(z_c + \frac{1}{M}\sum_{k=1}^{M} \nu_k \Big). \tag{4.91}$$

Define the variable

$$\theta = 1 + \langle \mathbf{N}, \mathbf{n}\rangle, \tag{4.92}$$

where $\theta \to 0$ when $\mathbf{n} \to -\mathbf{N}$. i.e. when the swarm speeds up or slows down in the negative direction of the field gradient. Taking the time derivative of θ yields

$$\dot{\theta} = \langle \mathbf{N}, \dot{\mathbf{n}}\rangle + \langle \mathbf{n}, \dot{\mathbf{N}}\rangle. \tag{4.93}$$

Using (4.70) for $\dot{\mathbf{n}}$ and substituting θ in (4.91), we obtain the reduced system

$$\begin{aligned} \dot{z}_c &= k_1 ||\nabla z_c|| (\theta - 1)(z_c + \frac{1}{M}\sum_{k=1}^{M} \nu_k), \\ \dot{\theta} &= \frac{k_1 \lambda^q ||\nabla z_c||}{\lambda^q - \lambda^n}\theta(\theta - 2) + \frac{k_1}{\lambda^q - \lambda^n}\Delta + \delta, \end{aligned} \tag{4.94}$$

where $\Delta = -\langle \mathbf{N}, \mathbf{q}\rangle \sum_{k=1}^{M} \nu_k \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q}\rangle$ is due to the nonlinear components of the field, and $\delta = \langle \mathbf{n}, \dot{\mathbf{N}}\rangle$ is viewed as an unknown input field disturbance.

Let $\dot{\mathbf{x}} = f(\mathbf{x}, \delta)$ where $\mathbf{x} = [z_c, \theta]^T$ and f is as defined by (4.94), and δ is the input field disturbance. Then we have following result for the reduced system.

Lemma 4.5.1. *Consider the reduced system (4.94) and suppose* $0 \leq \theta(0) \leq 1$. *Then the origin of the unforced system* $f(z_c, \theta, 0)$ *is asymptotically stable. Furthermore, suppose that* $\Big(k_1 ||\nabla z_c|| \lambda^q |\nu| + k_1 |\Delta| + (\lambda^q - \lambda^n) |\delta| \Big) / (\epsilon_1 k_1 ||\nabla z_c|| \lambda^q) < z_c(0)^2$, *then the origin of forced system* $f(z_c, \theta, \delta)$ *is input-to-state stable.*

Proof. Consider the domain $D_1 = \{[z_c, \theta]^T | z_c \geq 0, \theta \in [0, 1)\}$. Note that $z_c \geq 0$ holds

everywhere since by **Assumption 4.2.3** the field is positive. Additionally, $\theta \in [0, 1)$ implies that $\langle \mathbf{N}, \mathbf{n} \rangle < 0$. Let $V_1 : D_1 \to \mathbf{R}$ be a Lyapunov candidate function defined by

$$V_1 = \frac{1}{2} z_c^2 + \Big(\frac{\lambda^q - \lambda^n}{\lambda^q}\Big)\Big(\frac{\theta}{1-\theta}\Big), \tag{4.95}$$

where $V_1 = 0$ if and only if $[z_c, \theta]^T = [0, 0]^T$. Additionally, $V_1 \to \infty$ as $\theta \to 1$.

For the unforced system $f(\mathbf{x}, 0)$, we set $\delta = \langle \mathbf{n}, \dot{\mathbf{N}} \rangle = 0$. This implies that either $\mathbf{n} = -\mathbf{N}$, i.e. the desired equilibrium, or $\dot{\mathbf{N}} = 0$. But $\dot{\mathbf{N}} = \frac{d\nabla z_c}{dt} = 0$ if and only if the field gradient ∇z_c is linear, i.e. $\nu_k = 0$. Hence, $\delta = 0$ implies that $\Delta = 0$. Taking the derivative of V_1 and using (4.86) and (4.94), we obtain

$$\begin{aligned}
\dot{V}_1 &= z_c \dot{z}_c - \frac{1}{\lambda^q}\Big(\frac{\theta}{1-\theta}\Big)\dot{\lambda}^n + \Big(\frac{\lambda^q - \lambda^n}{\lambda^q}\Big)\frac{\dot{\theta}}{(1-\theta)^2} \\
&= -k_1 ||\nabla z_c|| \Big((1-\theta) z_c^2 + \frac{\theta(2-\theta)}{(1-\theta)^2} - 2\frac{\lambda^n}{\lambda^q}\theta\Big) \\
&= -k_1 ||\nabla z_c|| \Big((1-\theta) z_c^2 + \theta[\frac{2-\theta}{(1-\theta)^2} - 2\frac{\lambda^n}{\lambda^q}]\Big) \leq 0,
\end{aligned}$$

where we used the fact that since $\frac{2-\theta}{(1-\theta)^2} \geq 2$ and $2\frac{\lambda^n}{\lambda^q} \leq 2$, then $\frac{2-\theta}{(1-\theta)^2} - 2\frac{\lambda^n}{\lambda^q} \geq 0$. Since $\dot{V}_1 = 0$ if and only if $[z_c, \theta]^T = [0, 0]^T$, then the origin of the unforced system $f(\mathbf{x}, 0)$ is asymptotically stable. Additionally, $\dot{V}_1 \to -\infty$ as $\theta \to 1$. This along with the fact that $V_1 \to \infty$ whenever $\theta \to 1$, implies that $\mathbf{D}_1$ is a forward invariant set and thus trajectories start inside it will never go outside it.

For the forced system $f(z_c, \theta, \delta)$, let

$$W_1(z_c, \theta) = k_1 ||\nabla z_c|| \Big((1-\theta) z_c^2 + \theta[\frac{2-\theta}{(1-\theta)^2} - 2\frac{\lambda^n}{\lambda^q}]\Big). \tag{4.96}$$

Note that $W_1(z_c, \theta)$ is a continuous positive definite function. Additionally, $W_1 = 0$ if and

only if $[z_c, \theta]^T = [0, 0]^T$. Hence, for $\epsilon_1 \in (0, 1)$, we obtain

$$\dot{V}_1 \leq -(1 - \epsilon_1)W_1, \quad \forall |z_c| > \rho(|\delta|), \tag{4.97}$$

where $\rho(|\delta|)$ is as given by

$$\rho(|\delta|) = \Big(\frac{|\nu|}{\epsilon_1} + \frac{|\Delta|}{\epsilon_1 \lambda^q ||\nabla z_c||} + \frac{\lambda^q - \lambda^n}{\lambda^q} \frac{|\delta|}{\epsilon_1 k_1 ||\nabla z_c||}\Big)^{\frac{1}{2}}, \tag{4.98}$$

which is a class $\mathcal{K}$ function obtained to sufficiently ensure (4.97).Using the assumption that $\Big(k_1 ||\nabla z_c|| \lambda^q |\nu| + k_1 |\Delta| + (\lambda^q - \lambda^n) |\delta|\Big)/(\epsilon_1 k_1 ||\nabla z_c|| \lambda^q) < z_c(0)^2$, then the set $\{z_c | \rho(|\delta|) < z_c < z_c^2(0)\}$ is not empty. Moreover, using (4.96), $\dot{V}_1 \to -\infty$ as $\theta \to 1$. This along with the fact that $V_1 \to \infty$ whenever $\theta \to 1$, implies that $\mathbf{D}_1$ is a forward invariant set. Let $\alpha_1(|| [z_c, \theta]^T ||) = \alpha_2(|| [z_c, \theta]^T ||) = \frac{1}{2} z_c^2 + \frac{\lambda^q - \lambda^n}{\lambda^q} \frac{\theta}{1-\theta}$ which are class $\mathcal{K}_\infty$ functions that satisfy: $\alpha_1(|| [z_c, \theta]^T ||) \leq V_1([z_c, \theta]^T) \leq \alpha_2(|| [z_c, \theta]^T ||)$. Therefore, according to *Theorem 4.19* in [68], the origin of the forced system $f(z_c, \theta, \delta)$ is input-to-state stable. □

Remark 5. *Note that* $\nu = (1/M) \sum_k \nu_k = (1/M) \sum_k z_k - z(\mathbf{r}_c)$*, i.e. the difference between the average and the center values of the field, can be arbitrary small when* M *is large or when the field is almost linear around the center. Additionally,* $\Delta = -\langle \mathbf{N}, \mathbf{q} \rangle \sum_{k=1}^{M} \nu_k \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q} \rangle \leq \sum_{k=1}^{M} |\nu_k| \, ||\mathbf{r}_k - \mathbf{r}_c||$*, can also be be arbitrary small when the field is almost linear around the center or when the agents are close to each other. On the other hand,* $(\lambda^q - \lambda^n) |\delta|$ *vanishes either at the equilibrium or when* $\lambda^q \approx \lambda^n$*, i.e. the swarm is more spatially distributed. Consequently,* $\rho(|\delta|)$ *can be arbitrary small, however, when the swarm is close to the source, then* $|z_c| > \rho(|\delta|)$ *does not hold. Without a termination policy in the Algorithm, the swarm may pass the source. Fortunately,* ***Lemma 4.4.3*** *shows that the swarm is guaranteed to switch between* $\mathbf{n}$ *and* $\mathbf{q}$ *and hence the swarm steers back to the set* $\theta \in (0, 1)$.

The Boundary System

Define

$$\psi = 1 - \langle \mathbf{q}(t), \mathbf{q}(\tau) \rangle, \tag{4.99}$$

where $\psi \to 0$ when $\mathbf{q}(\tau) \to \mathbf{q}(t)$, i.e when the PCA learning algorithm converges to the exact eigenvector of the covariance matrix $\mathbf{C}(t)$. By the Chain rule, $\frac{d\mathbf{q}(t)}{d\tau} = \epsilon \frac{d\mathbf{q}(t)}{dt}$, and $\frac{d\mathbf{q}(\tau)}{dt} = \frac{1}{\epsilon} \frac{d\mathbf{q}(\tau)}{d\tau}$. Hence

$$\frac{d\psi}{d\tau} = \epsilon \frac{d\psi}{dt} = -\epsilon \langle \frac{d\mathbf{q}(t)}{dt}, \mathbf{q}(\tau) \rangle - \langle \mathbf{q}(t), \frac{d\mathbf{q}(\tau)}{d\tau} \rangle. \tag{4.100}$$

From (4.3), we obtain

$$\begin{aligned} \langle \mathbf{q}(t), \frac{d\mathbf{q}(\tau)}{d\tau} \rangle &= \langle \mathbf{q}(t), (\mathbf{I} - \mathbf{q}(\tau)\mathbf{q}^T(\tau)) \mathbf{C}(t) \mathbf{q}(\tau) \rangle \\ &= \langle \mathbf{q}(t), \mathbf{q}(\tau) \rangle \Big(\lambda^q - \langle \mathbf{q}(\tau), \mathbf{C}(t) \mathbf{q}(\tau) \rangle \Big). \end{aligned} \tag{4.101}$$

On the other hand, using (4.64), we obtain

$$\langle \frac{d\mathbf{q}(t)}{dt}, \mathbf{q}(\tau) \rangle = \frac{k_1}{\lambda^q - \lambda^n} \Big(||\nabla z_c|| \lambda^q \langle \mathbf{N}, \mathbf{q} \rangle + \sum_{k=1}^{M} \nu_k \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q} \rangle \Big) \langle \mathbf{n}, \mathbf{q}(\tau) \rangle. \tag{4.102}$$

Substituting (4.101) and (4.102) in (4.100), we obtain

$$\begin{aligned} \frac{d\psi}{d\tau} = \epsilon \frac{d\psi}{dt} &= -(1 - \psi) \Big(\lambda^q - \langle \mathbf{q}(\tau), \mathbf{C}(t) \mathbf{q}(\tau) \rangle \Big) \\ &\quad - \frac{\epsilon k_1}{\lambda^q - \lambda^n} \Big(||\nabla z_c|| \lambda^q \langle \mathbf{N}, \mathbf{q} \rangle + \sum_{k=1}^{M} \nu_k \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q} \rangle \Big) \langle \mathbf{n}, \mathbf{q}(\tau) \rangle. \end{aligned} \tag{4.103}$$

Setting $\epsilon = 0$ in (4.103), we obtain the boundary system

$$\frac{d\psi}{d\tau} = (1-\psi)(\langle \mathbf{q}(\tau), \mathbf{C}(t)\mathbf{q}(\tau)\rangle - \lambda^q). \tag{4.104}$$

Observe that in (4.104), $\mathbf{q}(t)$ and $\mathbf{C}(t)$ are constants with respect to the time scale τ. We have the following result for the boundary system

Lemma 4.5.2. *Consider (4.104). Suppose that at time $\tau = 0$, $\langle \mathbf{q}(t), \mathbf{q}(\tau)\rangle \in (0, 1]$. Then the origin of the boundary system is exponentially stable uniformly in $\mathbf{C}(t)$ and $\mathbf{q}(t)$.*

Proof. Let $D_2 = \{\psi \in \mathbb{R} | \psi \in [0, 1)\}$ which is equivalent to $\langle \mathbf{q}(t), \mathbf{q}(\tau)\rangle \geq 0$. Then let $V_2(\psi) : \mathbf{D}_2 \to \mathbb{R}$ be a Lyapunov candidate function defined by

$$V_2 = \frac{\psi}{1-\psi}, \tag{4.105}$$

where $V_2 \geq 0$ and $V_2 = 0$ if and only if $\psi = 0$. Furthermore, $V_2 \to \infty$ as $\psi \to 1$. Using (4.104) and the fact that $\langle \mathbf{q}(\tau), \mathbf{C}\mathbf{q}(\tau)\rangle \leq \lambda_{max}(\mathbf{C}) = \lambda^q$, we obtain

$$\frac{dV_2}{d\tau} = \frac{1}{(1-\psi)^2}\frac{d\psi}{d\tau} = \frac{1}{(1-\psi)}(\langle \mathbf{q}(\tau), \mathbf{C}\mathbf{q}(\tau)\rangle - \lambda^q) \leq 0, \tag{4.106}$$

where $\frac{dV_2}{d\tau} = 0$ if and only if $\langle \mathbf{q}(\tau), \mathbf{C}\mathbf{q}(\tau)\rangle = \lambda^q$, i.e. $\psi = 0$. Furthermore, we can write (4.106) as

$$\begin{aligned}
\frac{dV_2}{d\tau} &= \frac{\psi}{(1-\psi)}(\langle \mathbf{q}(\tau), \mathbf{C}\mathbf{q}(\tau)\rangle - \lambda^q) + \frac{1-\psi}{(1-\psi)}(\langle \mathbf{q}(\tau), \mathbf{C}\mathbf{q}(\tau)\rangle - \lambda^q) \\
&= -(\lambda^q - \langle \mathbf{q}(\tau), \mathbf{C}\mathbf{q}(\tau)\rangle)V_2 - (\lambda^q - \langle \mathbf{q}(\tau), \mathbf{C}\mathbf{q}(\tau)\rangle) \\
&\leq -(\lambda^q - \langle \mathbf{q}(\tau), \mathbf{C}\mathbf{q}(\tau)\rangle)V_2.
\end{aligned} \tag{4.107}$$

Consequently, the equilibrium $\psi = 0$ of (4.104) is exponentially stable. Then, according to *Definition 11.1* in [68], the equilibrium $\psi = 0$ of the boundary system (4.104) is exponentially stable, uniformly in $\mathbf{q}(t)$ and $\mathbf{C}(t)$. Note that, uniformly means the proof

holds for any $\mathbf{q}(t)$ and $\mathbf{C}(t)$ which are viewed as constants with respect to the time domain τ. Moreover, since $V_2 \to \infty$ whenever $\psi \to 1$, then $\mathbf{D}_2$ is a positively invariant set which implies that trajectories start inside it will stay there forever. □

The Coupled System

Define the coupled system $[\dot{z}_c, \dot{\theta}, \dot{\psi}] = h(\epsilon, z_c, \theta, \psi, \delta)$ where $\dot{z}_c$ and $\dot{\theta}$ are given by (4.94), and $\dot{\psi}$ is given by (4.103). The following theorem establishes a sufficient range for ϵ that generalizes **Lemma 4.5.1** and **Lemma 4.5.2** of the decoupled system $h(0, z_c, \theta, \psi, \delta) = f(z_c, \theta, \psi, \delta)$ to the coupled system $h(\epsilon, z_c, \theta, \psi, \delta)$.

Theorem 4.5.1. *Consider the coupled system given by (4.87) and (4.88) where $k_2 = 0$ and $z^d = 0$. Suppose that $\epsilon \leq \epsilon^*$ where*

$$\epsilon^* = \frac{\epsilon_2 \lambda^q (\lambda^q - \lambda^n)}{k_1 \epsilon_z \lambda^q + k_1 \sum_k \nu_k ||\mathbf{r}_k - \mathbf{r}_c||}. \tag{4.108}$$

If $0 \leq \theta(0) \leq 1$ and $0 \leq \psi(0) \leq 1$, then the origin of the unforced system $h(\epsilon, z_c, \theta, \psi, 0)$ is asymptotically stable. Furthermore, if $0 \leq \theta(0) \leq 1$, $0 \leq \psi(0) \leq 1$, and $\Big(k_1 ||\nabla z_c|| \lambda^q |\nu| + k_1 |\Delta| + (\lambda^q - \lambda^n) |\delta| \Big) / (\epsilon_1 k_1 ||\nabla z_c|| \lambda^q) < z_c(0)^2$, then the origin of forced system $h(\epsilon, z_c, \theta, \psi, \delta)$ is input-to-state stable.

Proof. Consider the domain $D = D_1 \cup D_2 = \{[z_c, \theta, \psi]^T | z_c \geq 0, \theta \in [0, 1), \psi \in [0, 1)\}$. Let $V : D \to \mathbb{R}$ be a Lyapunov candidate function for the overall system and defined as $V = V_1 + V_2$. Since $\epsilon \neq 0$, then using (4.103), we obtain

$$\dot{V}_2 = -\frac{1}{\epsilon} W_2 + Q_2, \tag{4.109}$$

where

$$W_2 = \frac{\lambda^q - \langle \mathbf{q}(\tau), \mathbf{C}(t)\mathbf{q}(\tau) \rangle}{1 - \psi} \tag{4.110}$$

is a continuous positive definite function in the domain D_2, and

$$Q_2 = -\frac{k_1}{(\lambda^q - \lambda^n)(1-\psi)^2}\Big(||\nabla z_c||\lambda^q\langle \mathbf{N}, \mathbf{q}\rangle + \sum_{k=1}^{M}\nu_k\langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q}\rangle\Big)\langle \mathbf{n}, \mathbf{q}(\tau)\rangle \quad (4.111)$$

is an indefinite function due to the mismatch between $\mathbf{q}(t)$ and $\mathbf{q}(\tau)$. Let $\epsilon_2 \in (0,1)$. Then using (4.97) and (4.109) we obtain

$$\dot{V} \leq -(1-\epsilon_2)(W_1 + \frac{1}{\epsilon}W_2), \quad \forall |z_c| > \rho(|\delta|), \epsilon \leq \epsilon^*, \quad (4.112)$$

where $\epsilon^* = \frac{\epsilon_2\lambda^q(\lambda^q-\lambda^n)}{k_1\epsilon_z\lambda^q + k_1\sum_k(\nu^{max}-\nu_k)||\mathbf{r}_k - \mathbf{r}_c||}$. Hence, the the results of the decoupled systems in **Lemma 4.5.1** and **Lemma 4.5.2** hold whenever $\epsilon \leq \epsilon^*$. This implies that the overall system is input-to-state stable. □

Note that if the higher order terms of the field are ignored, then ϵ^* reduces to $\epsilon^* = \frac{\epsilon_2(\lambda^q-\lambda^n)}{k_1||\nabla z_c||}$. Additionally, ϵ^* defines how long we should run the PCA flow (4.3) in the time scale τ for each instant of time t. Since $\tau = \frac{t-t_0}{\epsilon}$, then a small value of ϵ^* requires a longer PCA learning time τ, and vice versa.

4.5.2 Source Seeking with an Incomplete Graph

When the graph is incomplete, the implicit dynamics of the body frame are as defined in **Corollary** 4.4.1.

The Reduced System

Let $z_{c,i}$ be the field measurement at the local center $\mathbf{r}_{c,i}$. Define

$$\theta_i = 1 + \langle \mathbf{N}_i, \mathbf{n}_i\rangle, \quad (4.113)$$

where $\theta_i \to 0$ when $\mathbf{n}_i \to -\mathbf{N}_i$. i.e. when $\mathbf{n}_i$ converges to the negative direction of the local field gradient. Taking the time derivative of $z_{c,i}$ and θ, and using (4.52) for $\dot{\mathbf{n}}_i$, we

obtain the reduced system

$$\dot{z}_{c,i} = k_1||\nabla z_i||(\theta_i - 1)z_{c,i}, \quad i = 1, \cdots, M$$
$$\dot{\theta}_i = k_1||\nabla z_i||\frac{\lambda_i^q}{\lambda_i^q - \lambda_i^n}\theta_i(\theta_i - 2) + \beta_i, i = 1, \cdots, M, \tag{4.114}$$

where the input disturbance is defined as

$$\beta_i = \mathcal{E}_i\langle \mathbf{N}_i, \mathbf{q}_i\rangle + \delta_i, \quad \delta_i = \langle \mathbf{n}_i, \dot{\mathbf{N}}_i\rangle. \tag{4.115}$$

The term $\mathcal{E}_i$ accounts for the mismatch between the local PCA components, and the term δ_i accounts for how the local field gradient ∇z_i changes as the swarm moves from one location to another. Observe that $\beta_i = 0$ when $\theta_i = 0$. Additionally, $\mathcal{E}_i$ is determined by the graph structure which in the absence of a formation force in (4.4) is hardly controlled. The only hope to decrease $\mathcal{E}_i$ is to increase the connectivity of the static graph as much as possible, as shown in Fig. 4.5.

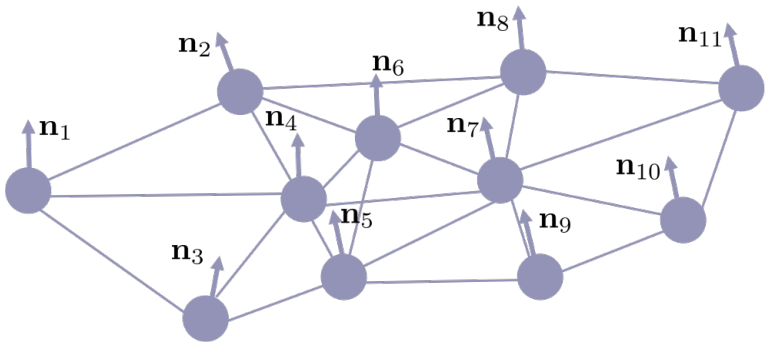


Figure 4.5: An incomplete graph of 11 agents. The lines represent the undirected edges, and the arrows represent the local PCA component $\mathbf{n}_i$.

Define

$$\mathbf{z} = [z_{c,1}, \cdots, z_{c,M}]^T, \quad \theta = [\theta_i, \cdots, \theta_M]^T, \quad \beta = [\beta_i, \cdots, \beta_M]^T. \tag{4.116}$$

Let

$$\dot{\mathbf{x}} = [f_1(\mathbf{x}_1, \beta_1), \cdots, f_M(\mathbf{x}_M, \beta_M)]^T, \tag{4.117}$$

where $\mathbf{x} = [\mathbf{z}, \theta]^T$ and f_i is as defined by (4.114). Then we obtain the following result for the reduced system.

Lemma 4.5.3. *Consider the reduced system (4.114) and suppose for all i, $0 \leq \theta_i(0) \leq 1$. Then the origin of the unforced system $f(\mathbf{x}, 0)$ is asymptotically stable. Furthermore, suppose that $(\sum_{i=1}^{M} |\mathcal{E}_i| + |\delta_i|)/(\epsilon_1 k_1 \min_i\{||\nabla z_i||\}) < \sum_{i=1}^{M} z_{c,i}(0)^2$, then the origin of forced system $f(\mathbf{x}, \delta)$ is input-to-state stable.*

Proof. Consider the domain $D_1 = \{[\mathbf{z}, \theta]^T | \forall i, \quad z_{c,i} \geq 0, \theta_i \in [0, 1)\}$. Let $V_1 : D_1 \to \mathbf{R}$ be a Lyapunov candidate function defined by

$$V_1 = \frac{1}{2}||\mathbf{z}||^2 + \sum_{i=1}^{M} \frac{\theta_i}{1 - \theta_i}, \tag{4.118}$$

where $V_1 = 0$ if and only if $[\mathbf{z}, \theta] = [\mathbf{0}, \mathbf{0}]$. Additionally, $V_1 \to \infty$ as $\theta \to \mathbf{1}$, where $\mathbf{0}$ and $\mathbf{1}$ are columns vectors of all zeros and ones, respectively.

For the unforced system $f(\mathbf{x}, 0)$, we set $\beta = \mathbf{0}$. Then, taking the derivative of V_1 and using (4.114), we obtain

$$\begin{aligned}
\dot{V_1} &= \sum_{i=1}^{M} \Big(z_{c,i} \dot{z}_{c,i} + \frac{\dot{\theta}_i}{(1 - \theta_i)^2} \Big) \\
&= -k_1 \sum_{i=1}^{M} ||\nabla z_i|| \Big((1 - \theta_i) z_{c,i}^2 + \frac{\lambda_i^q}{\lambda_i^q - \lambda_i^n} \frac{\theta_i (2 - \theta_i)}{(1 - \theta_i)^2} \Big) \\
&\leq 0,
\end{aligned} \tag{4.119}$$

where $\dot{V}_1 = 0$ if and only if $[\mathbf{z}, \theta] = [\mathbf{0}, \mathbf{0}]$. Then the origin of the unforced system $f(\mathbf{x}, \mathbf{0})$ is asymptotically stable. Additionally, $\dot{V}_1 \to -\infty$ as $\theta \to \mathbf{1}$. This along with the fact that

$V_1 \to \infty$ whenever $\theta \to \mathbf{1}$, implies that $\mathbf{D}_1$ is a forward invariant set and thus trajectories start inside it will never go outside it.

For the forced system $f(\mathbf{x}, \beta)$, let $\epsilon_1 \in (0, 1)$ be a constant, and let

$$W_1 = k_1 \sum_{i=1}^{M} ||\nabla z_i|| \Big((1-\theta_i) z_{c,i}^2 + \frac{\lambda_i^q}{\lambda_i^q - \lambda_i^n} \frac{\theta_i(2-\theta_i)}{(1-\theta_i)^2} \Big) \tag{4.120}$$

be a continuous positive definite function. Then, we obtain

$$\dot{V}_1 \leq -(1-\epsilon_1) W_1, \quad \forall ||\mathbf{z}|| > \rho(||\beta||), \tag{4.121}$$

where

$$\rho(||\beta|| = \Big(\frac{\sum_{i=1}^{M} |(\mathcal{E}_i| + |\delta_i|)}{\epsilon_1 k_1 \min_i \{||\nabla z_i||\}} \Big)^{\frac{1}{2}} \tag{4.122}$$

is a class $\mathcal{K}$ function. Using the assumption that $(\sum_{i=1}^{M} |\mathcal{E}_i| + |\delta_i|)/(\epsilon_1 k_1 \min_i \{||\nabla z_i||\}) < \sum_{i=1}^{M} z_{c,i}(0)^2$, then the set $\{z_c | \rho(|\beta|) < ||\mathbf{z}|| < ||\mathbf{z}(\mathbf{0})||\}$ is not empty. Moreover, using (4.120), $\dot{V}_1 \to -\infty$ as $\theta \to 1$. This along with the fact that $V_1 \to \infty$ if for any i, $\theta_i \to 1$, implies that $\mathbf{D}_1$ is a forward invariant set. Let $\alpha_1(||\mathbf{x}||) = \alpha_2(||\mathbf{x}||) = \frac{1}{2}||\mathbf{z}||^2 + \sum_{i=1}^{M} \frac{|\theta_i|}{1-|\theta_i|}$ which are class $\mathcal{K}_\infty$ functions that satisfy: $\alpha_1(||\mathbf{x}||) \leq V_1(\mathbf{x}) \leq \alpha_2(||\mathbf{x}||)$. Therefore, according to *Theorem 4.19* in [68], the origin of the forced system $f(\mathbf{x}, \beta)$ is input-to-state stable. □

The Boundary System

Define

$$\psi = \sum_{i=1}^{M} (1 - \langle \mathbf{q}_i(t), \mathbf{q}_i(\tau) \rangle), \tag{4.123}$$

where $\psi \rightarrow 0$ when $\mathbf{q}_i(\tau) \rightarrow \mathbf{q}_i(t)\ \forall i$, i.e when all the local PCA learning algorithms converge to the exact eigenvectors of the covariance matrices $\mathbf{C}_i(t)$. Following the same procedure in Section 4.5.1, using (4.3) and (4.53), we obtain

$$\begin{aligned}\frac{d\psi}{d\tau} = \epsilon\frac{d\psi}{dt} = &-\sum_{i=1}^{M}(1-\psi_i)\Big(\lambda_i^q - \langle \mathbf{q}_i(\tau), \mathbf{C}_i(t)\mathbf{q}_i(\tau)\rangle\Big) \\ &- \epsilon\sum_{i=1}^{M}\Big(k_1||\nabla z_i||\frac{\lambda_i^q}{\lambda_i^q - \lambda_i^n}\langle \mathbf{N}_i, \mathbf{q}_i\rangle + \mathcal{E}_i\Big)\langle \mathbf{n}_i, \mathbf{q}_i(\tau)\rangle. \end{aligned} \tag{4.124}$$

Setting $\epsilon = 0$ in (4.124), we obtain the boundary system

$$\frac{d\psi}{d\tau} = -\sum_{i=1}^{M}(1-\psi_i)\Big(\lambda_i^q - \langle \mathbf{q}_i(\tau), \mathbf{C}_i(t)\mathbf{q}_i(\tau)\rangle\Big). \tag{4.125}$$

The following lemma summarizes the convergence result of the boundary system.

Lemma 4.5.4. *Consider the boundary system (4.125). Suppose that for all agents at time $\tau = 0$, $\langle \mathbf{q}_i(t), \mathbf{q}_i(\tau)\rangle \in (0, 1]$. Then the origin of the boundary system is exponentially stable uniformly in all $\mathbf{C}_i(t)$ and $\mathbf{q}_i(t)$.*

Proof. Let $D_2 = \{\psi | \forall i, \psi_i \in [0, 1)\}$ where $\psi_i = 1 - \langle \mathbf{q}_i(t), \mathbf{q}_i(\tau)\rangle$. Then let $V_2(\psi) : \mathbf{D}_2 \rightarrow \mathbb{R}$ be a Lyapunov candidate function defined by

$$V_2 = \sum_{i=1}^{M}\frac{\psi_i}{1-\psi_i}, \tag{4.126}$$

where $V_2 \geq 0$ and $V_2 = 0$ if and only if $\psi_i = 0, \forall i$. Furthermore, $V_2 \rightarrow \infty$ as any $\psi_i \rightarrow 1$. Using (4.125) and the fact that $\langle \mathbf{q}_i(\tau), \mathbf{C}_i\mathbf{q}_i(\tau)\rangle \leq \lambda_{max}(\mathbf{C}_i) = \lambda_i^q$, we obtain

$$\frac{dV_2}{d\tau} = \sum_{i=1}^{M}\frac{1}{(1-\psi_i)}\Big(\langle \mathbf{q}_i(\tau), \mathbf{C}_i\mathbf{q}_i(\tau)\rangle - \lambda_i^q\Big) \leq 0. \tag{4.127}$$

where $\frac{dV_2}{d\tau} = 0$ if and only if $\forall i$, $\langle \mathbf{q}_i(\tau), \mathbf{C}_i\mathbf{q}_i(\tau)\rangle = \lambda_i^q$, i.e. $\psi_i = 0, \forall i$. Furthermore, from

(4.127)

$$\frac{dV_2}{d\tau} = -\sum_{i=1}^{M}(\lambda_i^q - \langle \mathbf{q}_i(\tau), \mathbf{C}_i\mathbf{q}_i(\tau)\rangle)V_2 - \sum_{i=1}^{M}(\lambda_i^q - \langle \mathbf{q}_i(\tau), \mathbf{C}_i\mathbf{q}_i(\tau)\rangle)$$
$$\leq -\sum_{i=1}^{M}(\lambda_i^q - \langle \mathbf{q}_i(\tau), \mathbf{C}_i\mathbf{q}_i(\tau)\rangle)V_2. \tag{4.128}$$

Consequently, the equilibrium $\psi = 0$ of (4.125) is exponentially stable. Then, according to *Definition 11.1* in [68], the equilibrium $\psi = 0$ of the boundary system (4.125) is exponentially stable, uniformly in all $\mathbf{q}_i(t)$ and $\mathbf{C}_i(t)$. Moreover, since $V_2 \to \infty$ whenever any $\psi_i \to 1$, then $\mathbf{D}_2$ is a forward invariant set which implies that trajectories start inside it will stay there forever. □

The Coupled System

Define the coupled system $[\dot{\mathbf{z}}, \dot{\theta}, \dot{\psi}] = h(\epsilon, \mathbf{z}, \theta, \psi, \beta)$ where $\dot{\mathbf{z}}$ and $\dot{\theta}$ are given by (4.114), and $\dot{\psi}$ is given by (4.124). The following theorem establishes a sufficient range for ϵ that generalizes **Lemma 4.5.3** and **Lemma 4.5.4** of the decoupled system $h(0, \mathbf{z}, \theta, \psi, \beta) = f(\mathbf{z}, \theta, \psi, \beta)$ to the coupled system $h(\epsilon, \mathbf{z}, \theta, \psi, \beta)$.

Theorem 4.5.2. *Consider the coupled system given by (4.87) and (4.88) where $k_2 \neq 0$ and $z^d \neq 0$. Suppose that $\epsilon \leq \epsilon^*$ where*

$$\epsilon^* = \frac{\epsilon_2}{k_1\epsilon_z}(\sum_{i=1}^{M}\lambda_i^q)(\sum_{i=1}^{M}(\frac{\lambda_i^q}{\lambda_i^q - \lambda_i^n} + \mathcal{E}_i))^{-1}. \tag{4.129}$$

If for all i, $0 \leq \theta_i(0) \leq 1$ and $0 \leq \psi_i(0) \leq 1$, then the origin of the unforced system $h(\epsilon, \mathbf{z}, \theta, \psi, 0)$ is asymptotically stable. Furthermore, if $0 \leq \theta_i(0) \leq 1$, $0 \leq \psi_i(0) \leq 1$, and $(\sum_{i=1}^{M}|\mathcal{E}_i| + |\delta_i|)/(\epsilon_1 k_1 \min_i\{||\nabla z_i||\}) < \sum_{i=1}^{M} z_{c,i}(0)^2$, then the origin of forced system $h(\epsilon, \mathbf{z}, \theta, \psi, \beta)$ is input-to-state stable.

Proof. Consider the domain $D = D_1 \cup D_2 =$ where $D_1 = \{[\mathbf{z}, \theta]^T | \forall i, z_{c,i} \geq 0, \theta_i \in [0, 1)\}$ and $D_2 = \{\psi | \forall i, \psi_i \in [0, 1)\}$. Let $V : D \to \mathbb{R}$ be a Lyapunov candidate function for the

coupled system and defined as $V = V_1 + V_2$. Since $\epsilon \neq 0$, then using (4.124), we obtain

$$\dot{V}_2 = -\frac{1}{\epsilon} W_2 + Q_2, \tag{4.130}$$

where

$$W_2 = \sum_{i=1}^{M} \frac{\lambda_i^q - \langle \mathbf{q}_i(\tau), \mathbf{C}_i(t)\mathbf{q}_i(\tau) \rangle}{1 - \psi_i}, \tag{4.131}$$

is a continuous positive definite function in the domain D_2, and

$$Q_2 = -\sum_{i=1}^{M} \frac{k_1 ||\nabla z_i|| \lambda_i^q \langle \mathbf{N}_i, \mathbf{q}_i \rangle + \mathcal{E}_i}{(\lambda_i^q - \lambda_i^n)(1 - \psi_i)^2} \tag{4.132}$$

is generally an indefinite function. Let $\epsilon_2 \in (0, 1)$. Then using (4.121) and (4.130) we obtain

$$\dot{V} \leq -(1 - \epsilon_2)(W_1 + \frac{1}{\epsilon} W_2), \quad \forall ||\mathbf{z}|| > \rho(||\beta||) \text{ and } \epsilon \leq \epsilon^*, \tag{4.133}$$

where

$$\rho(||\beta||) = \Big(\frac{\sum_{i=1}^{M} (|\mathcal{E}_i| + |\delta_i|)}{\epsilon_1 k_1 \min_i \{||\nabla z_i||\}} \Big)^{\frac{1}{2}}, \tag{4.134}$$

and

$$\epsilon^* = \frac{\epsilon_2}{k_1 \epsilon_z} (\sum_{i=1}^{M} \lambda_i^q)(\sum_{i=1}^{M} (\frac{\lambda_i^q}{\lambda_i^q - \lambda_i^n} + \mathcal{E}_i))^{-1}. \tag{4.135}$$

Hence, according to **Theorem 11.3** in [68], whenever $\epsilon \leq \epsilon^*$, then the origin of forced system $h(\epsilon, \mathbf{z}, \theta, \psi, \beta)$ is input-to-state stable. □

4.6 Simulation and Experimental Results

In this section, we validate the proposed model through computer simulation and physical experiments. We used two robotic platforms: the surface Georgia Tech Robotarium unicycle robots [22], and the air Georgia Tech Miniature Autonomous Blimps [**GTMAB**]. In what follows, we first present the source seeking results and then the level curve tracking results. In all simulations and experiment, we set $\epsilon = 0.01$ in the (4.88). This means that we run the PCA flow (4.3) for a time $\tau = \frac{dt}{\epsilon}$ where $dt = 0.01$ is the step time used to update (4.4).

4.6.1 Simulation Results

We simulated **Algorithm 1** in virtual scalar convex and non-convex $2D$ fields for swarms that have complete and incomplete static connectivity graphs. In all the simulations, we set $k_1 = 1$ in (4.4). Additionally, the source is located at the origin. In all of the following figure, bold blue discs represent the agents and the blue arrows indicate the direction $\mathbf{n}_i$, where they turned to red color at the end of the simulation. The lines connecting the agents represent the edges of the network and the pink paths represent the trajectories of the agents. The contour lines represent the level curves of the field. In Fig. 4.6, swarms of 4 agents

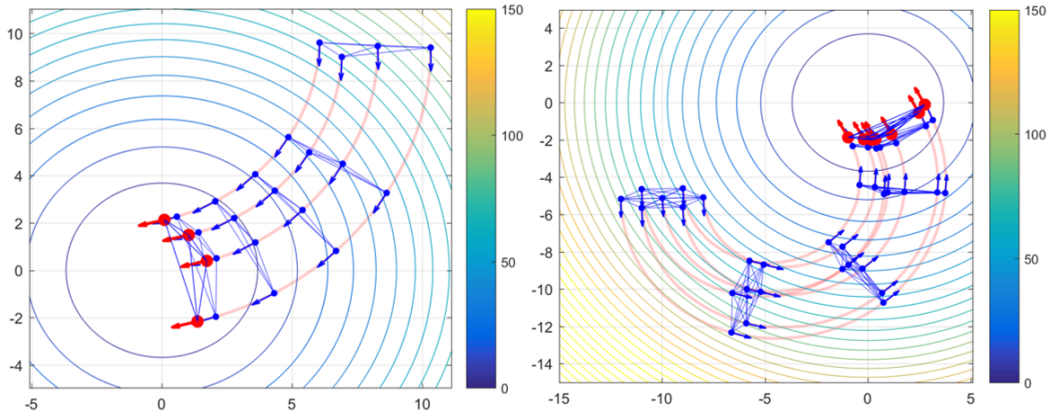


Figure 4.6: Swarms of 4 agents (left) and 7 agents (right) in complete graphs

(left) and 7 agents (right) in a complete graph are used to locate a convex field starting from

different initial positions. As predicted by **Theorem 4.5.1**, the two swarms successfully steered toward the source even though the 7-agent swarm was initially heading towards the positive direction of the field gradient. Additionally, as predicted by **Lemma 4.4.3**, the variance λ^q is constant while λ^n is varying.

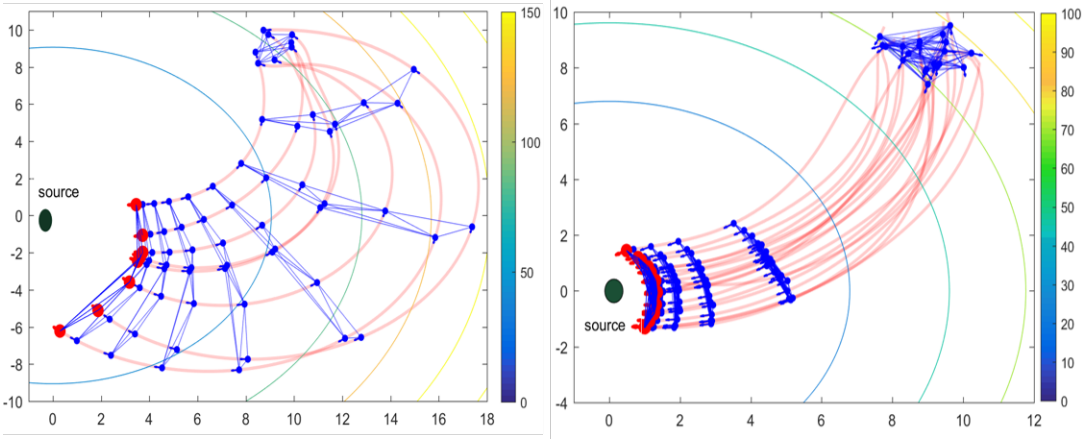


Figure 4.7: Swarms of 8 agents (left) and 20 agents (right) in an incomplete graphs

In the 8-agent swarm in the left of Fig. 4.7, the connectivity graph is incomplete. Hence each agent applies the PCA flow locally resulting in different $\mathbf{n}_i$, as it clear by the blue arrows at the initial time. Nevertheless, as predicted by **Theorem 4.5.2**, the swarm steers toward the source and each $\mathbf{n}_i$ converges to the negative direction of the local gradient $\mathbf{N}_i$. However, since initially for some agents $\langle \mathbf{n}_i, \mathbf{N}_i \rangle > 0$, the swarms disperse significantly. To save the connectivity of the graph when it is incomplete, in the 20-agent swarm in the right of Fig. 4.7, modify the control law (4.4) as

$$\mathbf{u}_i(t) = k_1(z_i(t) - z^d)\mathbf{n}_i(t) + k_2\mathbf{q}_i(t) + k_f\mathbf{v}_{i,\mathbf{q}}, \tag{4.136}$$

where k_f is a constant gain, and $v_{i,\mathbf{q}}$ is designed as follows

$$\mathbf{v}_{i,\mathbf{q}} = \sum_{j\in\mathcal{N}_i}(\langle \mathbf{r}_j - \mathbf{r}_i, \mathbf{q}_i\rangle - d_{ij})\langle \mathbf{r}_j - \mathbf{r}_i, \mathbf{q}_i\rangle\mathbf{q}_i, \tag{4.137}$$

which is to maintain a desired distance d_{ij} only along the $\mathbf{q}_i$ direction [10]. As shown in the right of Fig. 4.7, each $\mathbf{n}_i$ converges to $-\mathbf{N}_i$, but also, the agents in the swarm keep close to each other due to (4.137).

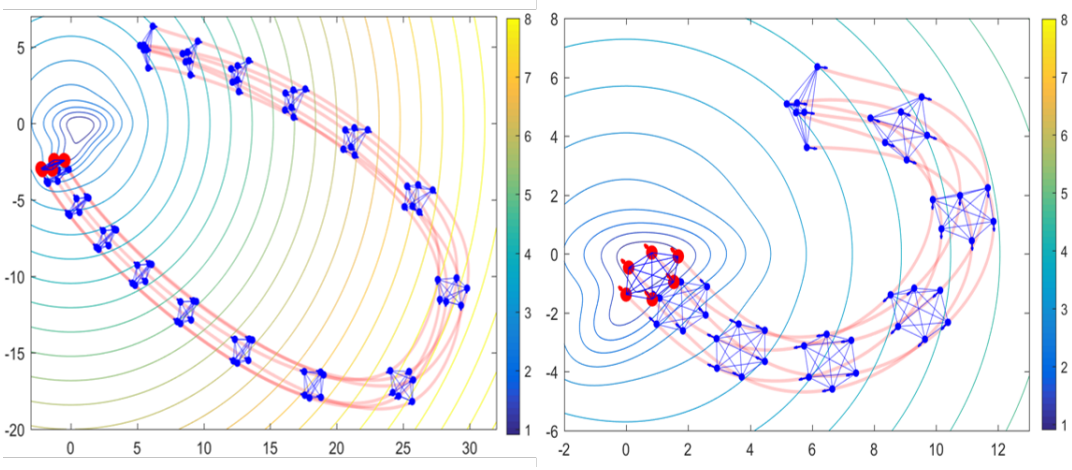


Figure 4.8: Complete network of 6 agents in a non convex field with partial formation (left) and full formation (right).

As suggested by (4.70), $\mathbf{n}$ changes faster when $(\lambda^q - \lambda^n)$ is small. To justify this, in Fig. 4.8, a swarm of 6 agents and a complete graph is simulated in a non-convex field. In the left, we added (4.136) to maintain a distance only along $\mathbf{q}$. However, in the right we maintain a distance along both directions by adding $\mathbf{v}_{i,\mathbf{n}}$ to (4.136), where $\mathbf{v}_{i,\mathbf{n}}$ is obtained by replacing $\mathbf{q}$ by $\mathbf{n}$ in (4.137). Although the two swarms start at the same location, since the one in the right maintained smaller $(\lambda^q - \lambda^n)$, it steered faster towards the source than the one in the left which took a long distance to turn. This intuitively reveals the effect of the different formation schemes. In particular, a swarm with a larger spatial distribution encodes more diverse information about the field and hence the swarm steers faster towards the source.

4.6.2 Experimental Results

We tested **Algorithm 1** using unicycle surface robots at the Georgia Teach Robotarium [22]. The diameter of each robot is about $0.05m$ and the dimensions of the experimental space are about $2m \times 3m$. Since these robots do not have light sensors, we used virtual

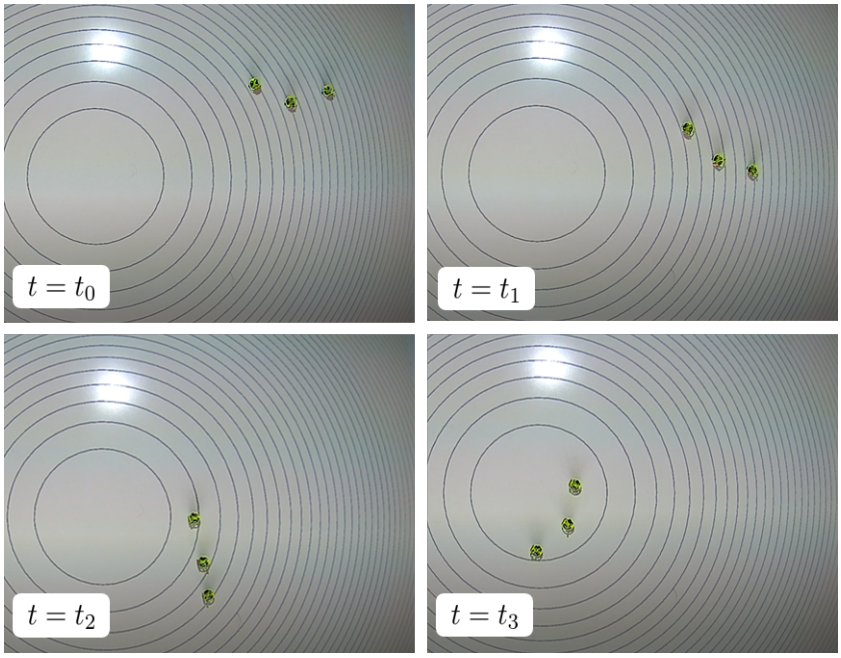


Figure 4.9: A 3-agent swarm in an incomplete graph

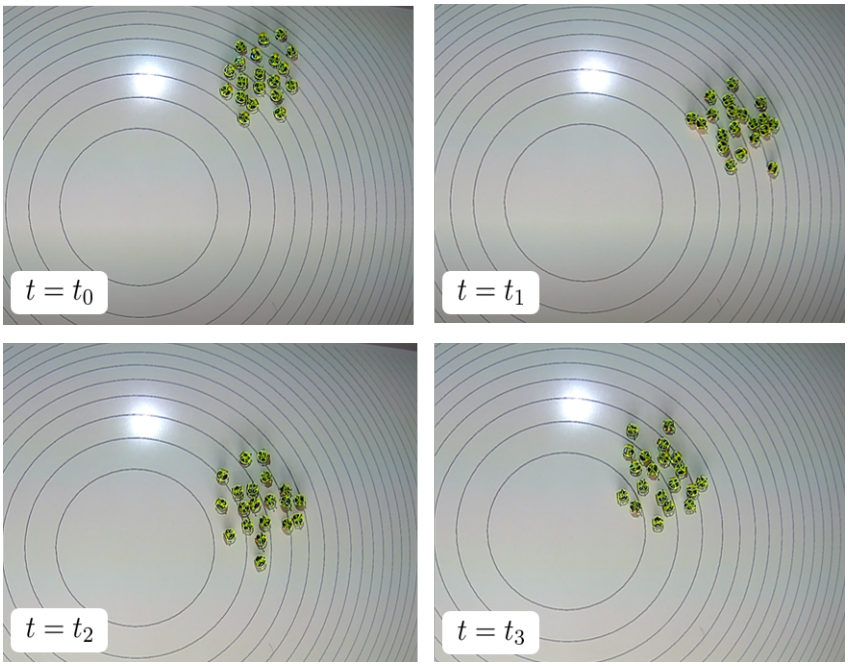


Figure 4.10: A 20-agent swarm in an incomplete graph

fields where we projected their level curves on the surface for a visualization purpose. We conducted three experiments using a convex field with 3 robots in Fig. 4.9, 20 robots in Fig. 4.10 and using a non-convex field with 6 robots in Fig. 4.11. These figures are

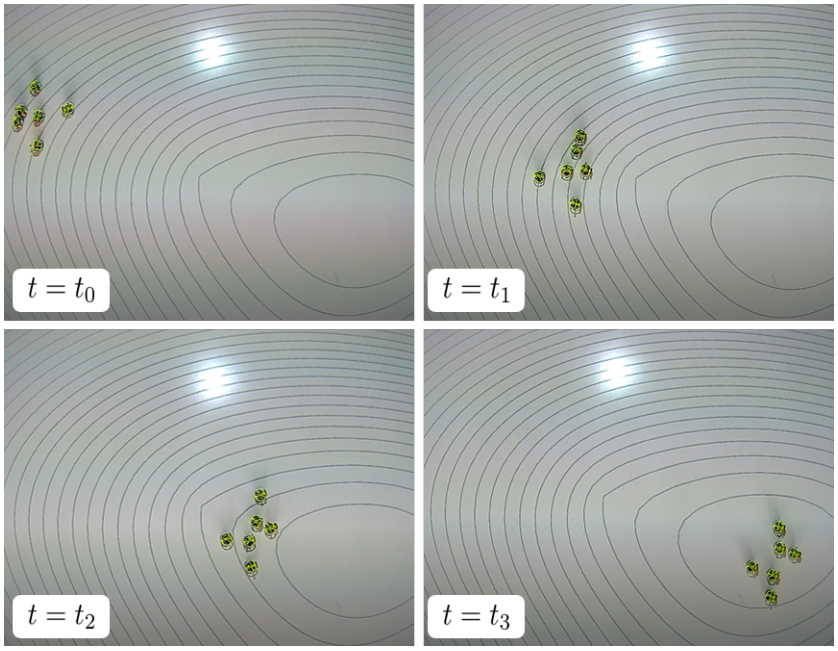


Figure 4.11: A 6-agent swarm in a complete graph and non-convex field

snapshots of the robots navigation from the start to end times. Despite the lack of a real field, these experiments reflect the successful performance of the proposed model when implemented in real robots.

Alternatively, we installed light sensors in the Georgia Tech Miniature Blimps [23] and then used a real light sensor. The diameter of each blimp is about $0.7m$ and the dimensions of the experimental space are about $4m \times 4m$. To make the minimum at the source, we inverted the field by using $\frac{1}{z_i}$ instead of z_i. Snapshots of two experiments are shown in Fig. 4.12 and Fig. 4.13, where initially in the former $\langle \mathbf{N}, \mathbf{n} \rangle < 0$, and in the latter $\langle \mathbf{N}, \mathbf{n} \rangle > 0$. Additionally, we presented in Fig. 4.14 and Fig. 4.15 trajectories of four different experiments. The trajectories are colored based on the light intensity where the black diamond is the source location and the red discs are the starting locations. Despite many messing measurements, the blimps are able to locate the source.

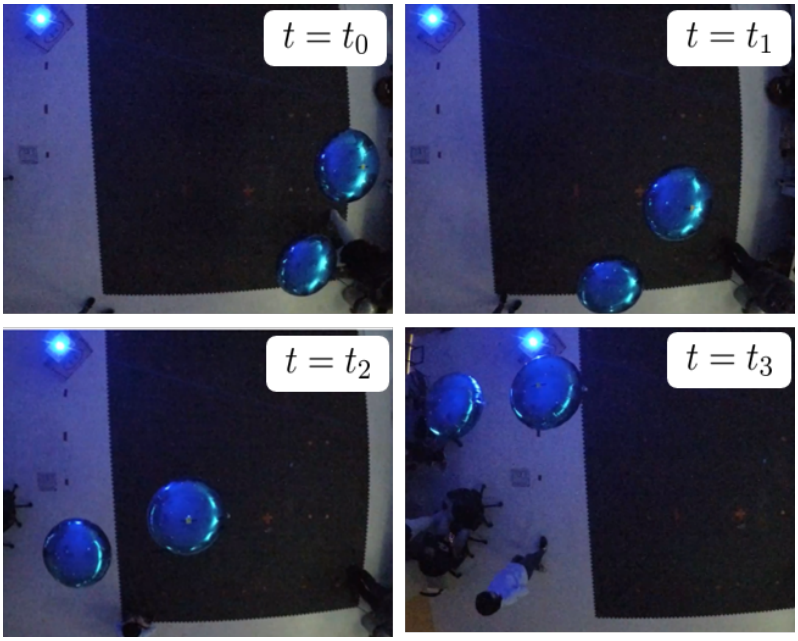


Figure 4.12: The two blimps initially have $\langle \mathbf{N}, \mathbf{n} \rangle < 0$.

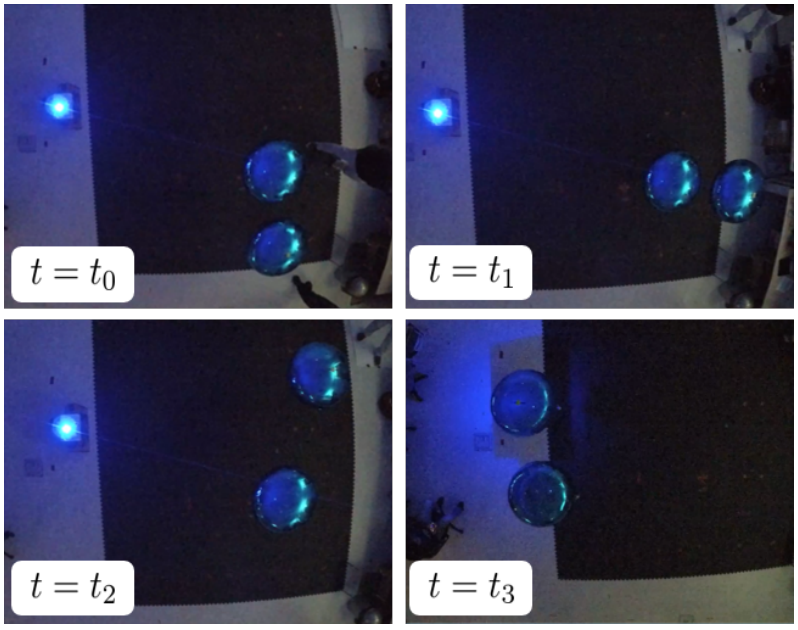


Figure 4.13: The two blimps initially have $\langle \mathbf{N}, \mathbf{n} \rangle > 0$.

4.7 Conclusion

In this chapter, we proposed a Multi-Layer control model composed of an interplay of distributed algorithms for perception and swarming. This enabled swarms of various sizes

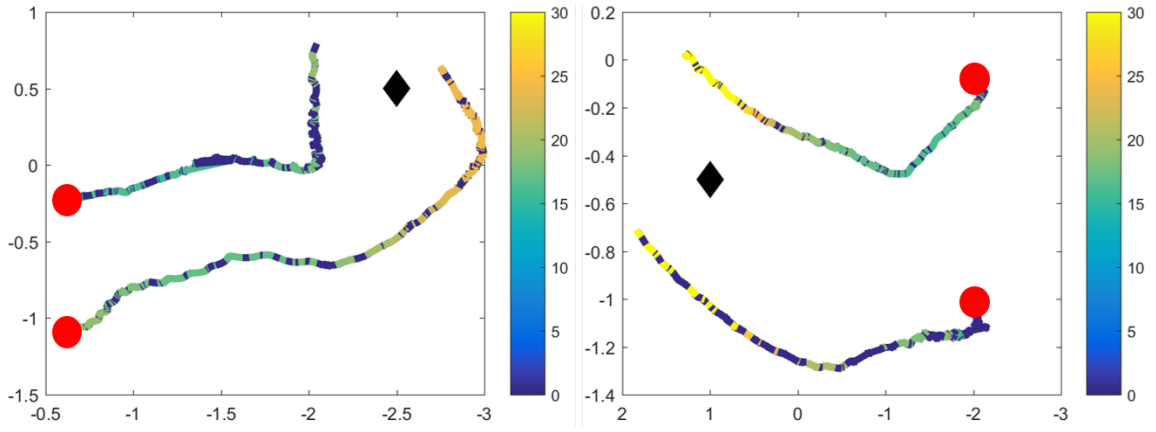


Figure 4.14: Trajectories of two experiments colored based on the light intensity where initially $\langle \mathbf{N}, \mathbf{n} \rangle < 0$.

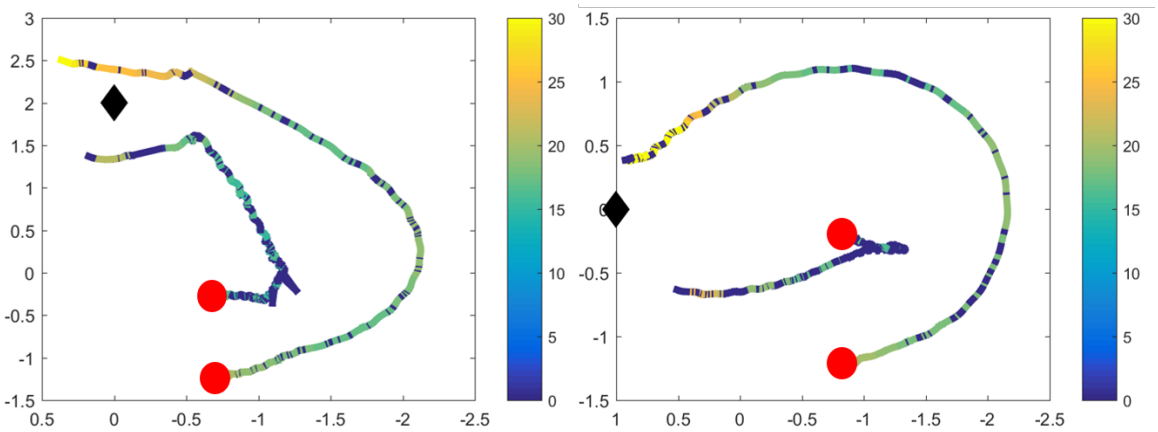


Figure 4.15: Trajectories of two experiments colored based the light intensity where initially $\langle \mathbf{N}, \mathbf{n} \rangle > 0$.

and graph structures to perform collective source seeking of scalar fields without the need to explicitly estimate the field gradient or explicitly share measurements among the agents. We obtained several stability results in a singular perturbation framework justifying the robustness and convergence of the algorithms. The simulation and experimental results suggest the efficiency and generality of the proposed model. To make the swarming control law more vigorous to incomplete graphs, in Chapter 6, we added a velocity alignment term to the swarming control law. Additionally, the implicit communication of the field measurements motivates us to use the model to study information propagation in biological swarms as well as designing new techniques for robotic swarms.

CHAPTER 5
INTEGRATING PCA LEARNING ALGORITHM FOR A DISTRIBUTED LEVEL CURVE TRACKING

5.1 Introduction

An important swarm robotics problem is to deploy multiple robots to track a desired level curve of a scalar field. The source is the location where the field has a maximum or minimum value. A level curve of a scalar field with a non-vanishing gradient is a curve consisting of all points where the field has the same value. The field can represent an environmental characteristic including but not limited to, a chemical concentration, a light intensity, or a temperature value. This problem has various applications ranging from monitoring environmental characteristics to exploring and establishing hazard boundaries, and to search and rescue tasks, just to name a few [6, 7].

In this chapter, we propose to use the Multi-Layer control model developed in Chapter 4, where we modify the control law in the swarming layer to achieve the level curve tracking behavior. Remarkably, through the interplay of the learning and swarming algorithms, swarms of various sizes and graph structures are able to perform collective level curve tracking of scalar fields without the need to explicitly estimate the field gradient or explicitly share measurements among the agents.

The different multi-agent control laws developed in the literature for curve tracking generally incorporate a mixture of knowledge or estimation of field gradient and Hessian estimation and sliding-mode control. Most of the aforementioned control strategies either rely on sharing measurements via communication channels, require maintaining specific spatial formations, or apply only to certain sizes and structures of the interacting graphs. The dependence on the exchange of data through a communication channel is a hard re-

quirement that might be undesired especially in applications with severe limitations such as underwater robotics.

In [71, 72] the field gradient is assumed to be known and then a control law is designed such that the agents move perpendicular to the gradient. Alternatively, the algorithms in [39] and [40] rely on communicating field measurements and maintaining prescribed formations to estimate the field gradient. Independent of gradient estimation, an algorithm is designed in [41] for a 2-agent system, but, it requires communicating field measurements. A discontinuous sliding mode control law is designed for a multi-agent system in [42] and [43] which is independent of both gradient estimation and measurements communication. However, each agent is able to track the level curve individually while the interaction with the other agents is mainly to force the agents to spread across the level curve.

The novel concept in this chapter is in integrating a PCA learning algorithm [13, 14, 15] in the perception layer of the Multi-Layer model through which each agent locally obtains a body frame. This time-varying body frame is then used in the swarming layer by each agent to modulate its motion based only on its instantaneous measurement of the field. The swarming control law is general in that by simple variations of the controller gains, the swarm exhibits different behaviors of source seeking and level curve tracking.

Similar to the source seeking algorithm of Chapter 4, the successful elimination of the challenging requirement of explicit estimation and communication is attributed to the locally computed PCA-based body frame and the design of the control law. We also overcome the difficulty of the multi-time-scale nature in the convergence analysis by incorporating a singular perturbation framework where the dynamics of the PCA learning and swarming control are viewed as rapidly decaying and slowly varying dynamics, respectively.

This chapter includes four main contributions [11, 9]. The first one is exploiting the Multi-Layer model to design an innovative solution to the level curve tracking problem. The second one is proving the boundedness of the spatial variances of the swarm under

complete graphs which renders implicit connectivity-maintenance. The third one is obtaining input-to-state stability results reflecting robust convergence to the desired level curve under complete graphs. The last one is validating the proposed model for various level curve tracking behaviors through simulations and experiments. The experiments are conducted using the Georgia Teach Robotarium [22] and the Georgia Teach Miniature Blimps [23].

The rest of this chapter is organized as follows. The problem is formulated in Section 5.2. Then the PCA-based body frame, the design of the control law and the strategy algorithm are presented in Section 5.3. In Section 5.4, we derive the implicit dynamics of the body frame and show how the field measurements are indirectly propagated. Then, stability analysis and simulation and experimental results are given in Section 5.5 and Section 5.6 respectively. Finally concluding remarks and suggestions of future work are provided in Section 5.7.

5.2 Problem Formulation

5.2.1 Preliminaries

Consider a swarm of M agents described by an undirected visibility graph, $\mathcal{G} \subseteq \mathcal{V} \times \mathcal{E}$ where $\mathcal{V}$ is the set of all agents, and $\mathcal{E}$ is the set of all edges. An undirected edge $(i, j) \in \mathcal{E}$ exists if both agents can sense the relative positions of each other. A graph is connected if for each $i, j \in \mathcal{V}$, there exists a sequence of edges connecting the i^{th} and j^{th} agents. If each agent shares an edge with all other agents, then the graph is complete, otherwise, it is incomplete. The neighbor set of i is defined by $\mathcal{N}_i = \{j|(i, j) \in \mathcal{E}\}$. Additionally, if for each agent $\mathcal{N}_i$ is fixed, the graph is static, otherwise, it is dynamic. We consider the following assumption about the graph.

Assumption 5.2.1. *$\mathcal{G}$ is static, undirected and connected.*

This assumption is to simplify the convergence analysis. However, the design in Sec-

tion 5.3 is applicable to a broader class of graphs which will be supported by simulation results. Additionally, we will show in Section 5.4 that connectivity is implicitly guaranteed when the graph is complete.

Let $\mathbf{r}_i \in \mathbb{R}^2$ be the position of the i^{th} agent in a 2D space. We require

Assumption 5.2.2. *each i^{th} agent to know the relative positions $\mathbf{r}_j - \mathbf{r}_i$ of all its neighbors, $j \in \mathcal{N}_i$.*

In practice, robots can be equipped with sensors to measure the relative positions of their neighbors with respect to their local frame, which is less challenging than requiring the global positions [5].

Furthermore, suppose each agent measures a scalar field value $z(\mathbf{r}_i) \in \mathbb{R}$ at its current position $\mathbf{r}_i(t)$, where the field is assumed to be

Assumption 5.2.3. *smooth, time invariant and bounded, i.e. $0 \leq z_{min} \leq z(\mathbf{r}_i) \leq z_{max}$, and has a unique minimum at the source location $\mathbf{r}_0$ where $z(\mathbf{r}_0) = z_{min}$.*

This assumption does not limit the proposed strategy as non-smooth fields may be transformed into smooth fields using, for example, Stochastic models as in [31].

Consider $z^d \in \mathbb{R}$ to be a desired level curve field value, where a level curve is the set $\{\mathbf{r} | z(\mathbf{r}) = z^d, \forall r \in \mathbb{R}^2\}$. Finally, let the velocity of each agent be described by

$$\dot{\mathbf{r}}_i = \frac{d\mathbf{r}_i}{dt} = \mathbf{u}_i, \quad i = 1, \cdots, M, \tag{5.1}$$

where $\mathbf{u}_i = \mathbf{u}_i(z^d, z_i(t), \{\mathbf{r}_j(t) - \mathbf{r}_i(t)\}_{j \in \mathcal{N}_i})$ is a control law that depends only on the desired level curve value z^d, current field measurement $z_i(t)$, and relative positions of the neighboring agents $\{\mathbf{r}_j(t) - \mathbf{r}_i(t)\}_{j \in \mathcal{N}_i}$.

5.2.2 Problem Statement

The problem to solve is to design the local control law $\mathbf{u}_i$ in (5.2), such that the swarm autonomously steers toward either the source location $\mathbf{r}_0$, or the desired level curve $\{\mathbf{r} | z(\mathbf{r}) =$

$z^d, \forall r \in \mathbb{R}^2\}$, and keeps tracking it. A challenging requirement we consider is to solve this problem without explicitly estimating the field gradient and without explicitly communicating field measurements among the agents.

5.3 The Distributed Control Law

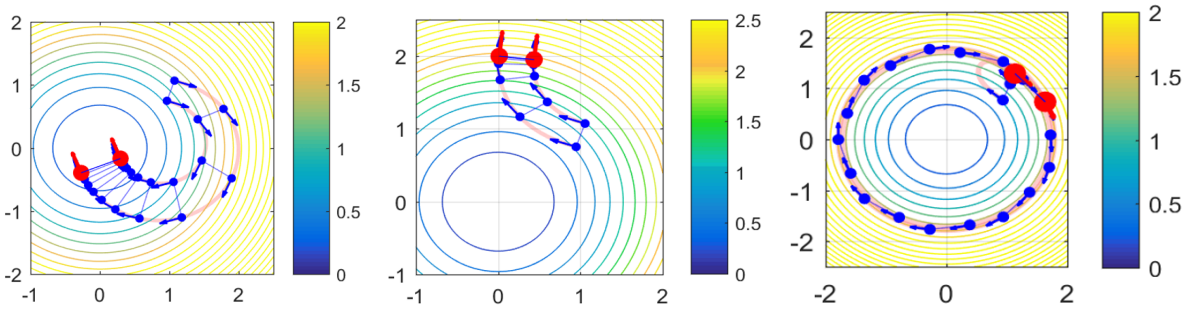
Given the body frame $(\mathbf{q}_i(t), \mathbf{n}_i(t))$ obtained by (4.3) in **Section** 4.3, we propose control law

$$\mathbf{u}_i(t) = k_1(z_i(t) - z^d)\mathbf{n}_i(t) + k_2\mathbf{q}_i(t), \tag{5.2}$$

where $z_i(t)$ and z^d are measured and desired field values, respectively. The parameters $k_1, k_2 \in \mathbb{R}$ are positive gains. To intuitively explain the control law (5.2), we simulate it in Fig. 5.1 for a 2-agent system in a scalar field. In this example, $\mathbf{q}_i$ is along the line-of-sight between the two agents, and $\mathbf{n}_i$ is perpendicular to the line-of-sight. When $z^d = 0$ and $k_2 = 0$, the control law (5.2) reduces to the source seeking control law (4.4). Namely, each agent i speeds up or slows down along the direction $\mathbf{n}_i$ depending on the local field measurement $z_i(t)$. Hence, the two agents move in the same direction with different speeds which steers them towards the minimum of the field.

On the other hand, when $z_d \neq 0$ and $k_2 = 0$, the 2-agent system steers toward the level curve $\{\mathbf{r}|z(\mathbf{r}_1) = z(\mathbf{r}_2) = z^d\}$. Finally, when $z_d \neq 0$ and $k_2 \neq 0$, then the first term $k_1(z_i(t) - z_d)\mathbf{n}_i(t)$ steers the 2-agent system toward the desired level curve, while the second term $k_2\mathbf{q}_i(t)$ moves the swarm along the level curve. Note that the first term changes its sign as the sign of $(z_i(t) - z_d)$ changes, which stabilize the agents at the level curve. Additionally, the gains k_1 and k_2 determine the tracking speed and accuracy. In particular, small k_2 compared to k_1 leads to slow tracking, but high accuracy, and vice versa.

A pseudocode description for the source seeking and level curve tracking strategy is given in Algorithm 2. Remarkably, all the steps in the algorithm are locally computed



(a) $z_d = 0$, $k_2 = 0$ (b) $z_d \neq 0$, $k_2 = 0$ (c) $z_d \neq 0$, $k_2 \neq 0$

Figure 5.1: The blue arrows are the velocities which turn red at the end time. The circular curves are the field level curves.

without explicit communication of any values.

Algorithm 2 The Distributed Source Seeking and Level Curve Tracking Strategy

```
loop
    for each agent do
        observe neighbor positions
        compute covariance matrix using (4.2)
        compute principal components using (4.3)
        update motion using (5.2)
    end for
end loop
```

5.4 The System Dynamics

The PCA flow system (4.3) describes the dynamics of the PCA components $(\mathbf{n}_i(t), \mathbf{q}_i(t))$ of a given covariance matrix $\mathbf{C}_i(t)$ only in the learning time scale τ. However, to analyze the convergence of the system, we also need the dynamics of the PCA components $(\mathbf{n}_i(t), \mathbf{q}_i(t))$ in the swarming time scale t. In this section, we derive these implicit dynamics for the general incomplete graph first, and then for the special complete graph. Additionally, at the end of this section, we show that the spatial variances of the swarm are bounded when the graph is complete.

Let $\mathcal{N}_i$ be the neighborhood set of agent i, and $\mathcal{H}_i = \mathcal{N}_i \bigcup \{i\}$, and $M_i = |\mathcal{N}_i| + 1$. The

covariance matrix seen by agent i is given by

$$\mathbf{C}_i = \sum_{k \in \mathcal{H}_i} (\mathbf{r}_k - \mathbf{r}_{c,i})(\mathbf{r}_k - \mathbf{r}_{c,i})^T, \tag{5.3}$$

where $\mathbf{r}_{c,i} = \frac{1}{M_i} \sum_{l \in \mathcal{H}_i} \mathbf{r}_l$ is the center of mass as seen by agent i. Using the same procedure in **Section** 4, we obtain

$$\dot{\mathbf{n}}_i = -\frac{1}{\lambda_i^q - \lambda_i^n} \langle \mathbf{q}_i, \dot{\mathbf{C}}_i \mathbf{n}_i \rangle \mathbf{q}_i \tag{5.4}$$

$$\dot{\mathbf{q}}_i = \frac{1}{\lambda_i^q - \lambda_i^n} \langle \mathbf{q}_i, \dot{\mathbf{C}}_i \mathbf{n}_i \rangle \mathbf{n}_i \tag{5.5}$$

Define $z_i^d = z_i - z_d$. We then prove the following result.

Lemma 5.4.1. *Using the motion dynamics (5.1) along with the PCA flow (4.3) and the control law (5.2), the implicit dynamics of the body frame for level curve tracking with general incomplete graphs are*

$$\dot{\mathbf{n}}_i = -k_1 \frac{1}{\lambda_i^q - \lambda_i^n} \mathbf{w}_i^T \mathbf{q}_i \mathbf{q}_i - \frac{1}{\lambda_i^q - \lambda_i^n} \sigma_i \mathbf{q}_i, \tag{5.6}$$

$$\dot{\mathbf{q}}_i = k_1 \frac{1}{\lambda_i^q - \lambda_i^n} \mathbf{w}_i^T \mathbf{q}_i \mathbf{n}_i + \frac{1}{\lambda_i^q - \lambda_i^n} \sigma_i \mathbf{n}_i, \tag{5.7}$$

where

$$\mathbf{w}_i = \sum_{k \in \mathcal{H}_i} \left(z_k^d \langle \mathbf{n}_k, \mathbf{n}_i \rangle - z_{c,i}^d \right) (\mathbf{r}_k - \mathbf{r}_{c,i}). \tag{5.8}$$

and

$$\sigma_i = k_1 \sum_{k\in\mathcal{H}_i} \Big(z_k^d \langle \mathbf{n}_k, \mathbf{q}_i\rangle - z_{c,i}\Big) \langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{n}_i\rangle + k_2 \sum_{k\in\mathcal{H}_i} \Big[\langle \mathbf{q}_k, \mathbf{n}_i\rangle \langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{q}_i\rangle + \langle \mathbf{q}_k, \mathbf{q}_i\rangle \langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{n}_i\rangle \Big]. \tag{5.9}$$

Proof. Applying time derivative on (5.3)

$$\dot{\mathbf{C}}_i = \sum_{k\in\mathcal{H}_i} [(\dot{\mathbf{r}}_k - \dot{\mathbf{r}}_{c,i})(\mathbf{r}_k - \mathbf{r}_{c,i})^T + (\mathbf{r}_k - \mathbf{r}_{c,i})(\dot{\mathbf{r}}_k - \dot{\mathbf{r}}_{c,i})^T]. \tag{5.10}$$

Using the control law (5.2), and letting $z_i^d = z_i - z_d$, we obtain

$$\begin{aligned} \dot{\mathbf{r}}_k - \dot{\mathbf{r}}_{c,i} &= k_1 z_k^d \mathbf{n}_k + k_2 \mathbf{q}_k - \frac{1}{M_i} \sum_{l\in\mathcal{H}_i} [k_1 z_l^d \mathbf{n}_l + k_2 \mathbf{q}_l] \\ &= \frac{M_i - 1}{M_i} [k_1 z_k^d \mathbf{n}_k + k_2 \mathbf{q}_k] - \frac{1}{M_i} \sum_{j\in\mathcal{N}_i} [k_1 z_j^d \mathbf{n}_j + k_2 \mathbf{q}_j] \\ &= \frac{1}{M_i} \sum_{j\in\mathcal{N}_i} \left[k_1 (z_k^d \mathbf{n}_k - z_j^d \mathbf{n}_j) + k_2 (\mathbf{q}_k - \mathbf{q}_j) \right]. \end{aligned} \tag{5.11}$$

Substituting (5.11) in (5.10)

$$\begin{aligned} \dot{\mathbf{C}}_i = &\frac{1}{M_i} \sum_{k\in\mathcal{H}_i} \sum_{j\in\mathcal{N}_i} \Big[(k_1 (z_k^d \mathbf{n}_k - z_j^d \mathbf{n}_j) + k_2 (\mathbf{q}_k - \mathbf{q}_j))(\mathbf{r}_k - \mathbf{r}_{c,i})^T \Big] \\ &+ \frac{1}{M_i} \sum_{k\in\mathcal{H}_i} \sum_{j\in\mathcal{N}_i} \Big[(\mathbf{r}_k - \mathbf{r}_{c,i})(k_1 (z_k^d \mathbf{n}_k - z_j^d \mathbf{n}_j) + k_2 (\mathbf{q}_k - \mathbf{q}_j))^T \Big]. \end{aligned} \tag{5.12}$$

Using (5.12), we obtain

$$\begin{aligned} \langle \mathbf{n}_i, \dot{\mathbf{C}}_i \mathbf{q}_i \rangle = &\frac{1}{M_i} \sum_{k\in\mathcal{H}_i} \sum_{j\in\mathcal{N}_i} \Big[k_1 (z_k^d \langle \mathbf{n}_k, \mathbf{n}_i\rangle - z_j^d \langle \mathbf{n}_j, \mathbf{n}_i\rangle) + k_2 \langle \mathbf{q}_k - \mathbf{q}_j, \mathbf{n}_i\rangle \Big] \langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{q}_i\rangle \\ &+ \frac{1}{M_i} \sum_{k\in\mathcal{H}_i} \sum_{j\in\mathcal{N}_i} \Big[k_1 (z_k^d \langle \mathbf{n}_k, \mathbf{q}_i\rangle - z_j^d \langle \mathbf{n}_j, \mathbf{q}_i\rangle) + k_2 \langle \mathbf{q}_k - \mathbf{q}_j, \mathbf{q}_i\rangle \Big] \langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{n}_i\rangle \end{aligned} \tag{5.13}$$

But

$$\frac{1}{M_i}\sum_{j\in\mathcal{N}_i}(z_k^d\langle \mathbf{n}_k,\mathbf{n}_i\rangle - z_j^d\langle \mathbf{n}_j,\mathbf{n}_i\rangle) = z_k^d\langle \mathbf{n}_k,\mathbf{n}_i\rangle - \frac{1}{M_i}\sum_{l\in\mathcal{H}_i} z_l^d\langle \mathbf{n}_l,\mathbf{n}_i\rangle. \tag{5.14}$$

Define $\tilde{z}_{a,i}^d$ such that

$$\frac{1}{M_i}\sum_{l\in\mathcal{H}_i} z_l^d\langle \mathbf{n}_l,\mathbf{n}_i\rangle = \tilde{z}_{a,i}^d\langle \mathbf{n}_k,\mathbf{n}_i\rangle. \tag{5.15}$$

Hence

$$z_k^d\langle \mathbf{n}_k,\mathbf{n}_i\rangle - \frac{1}{M_i}\sum_{l\in\mathcal{H}_i} z_l^d\langle \mathbf{n}_l,\mathbf{n}_i\rangle = (z_k^d - \tilde{z}_{a,i}^d)\langle \mathbf{n}_k,\mathbf{n}_i\rangle. \tag{5.16}$$

Since

$$\begin{aligned} z_k^d\langle \mathbf{n}_k,\mathbf{n}_i\rangle - \frac{1}{M_i}\sum_{l\in\mathcal{H}_i} z_l^d\langle \mathbf{n}_l,\mathbf{n}_i\rangle &= \frac{M_i-1}{M_i} z_k^d\langle \mathbf{n}_k,\mathbf{n}_i\rangle - \frac{1}{M_i}\sum_{j\in\mathcal{N}_i} z_j^d\langle \mathbf{n}_j,\mathbf{n}_i\rangle \\ &= \frac{1}{M_i}\sum_{j\in\mathcal{N}_i}(z_k^d\langle \mathbf{n}_k,\mathbf{n}_i\rangle - z_j^d\langle \mathbf{n}_j,\mathbf{n}_i\rangle), \end{aligned} \tag{5.17}$$

then

$$\frac{1}{M_i}\sum_{j\in\mathcal{N}_i}(z_k^d\langle \mathbf{n}_k,\mathbf{n}_i\rangle - z_j^d\langle \mathbf{n}_j,\mathbf{n}_i\rangle) = (z_k^d - \tilde{z}_{a,i}^d)\langle \mathbf{n}_k,\mathbf{n}_i\rangle. \tag{5.18}$$

Similarly, since

$$\begin{aligned} z_k^d\langle \mathbf{n}_k,\mathbf{q}_i\rangle - \frac{1}{M_i}\sum_{l\in\mathcal{H}_i} z_l^d\langle \mathbf{n}_l,\mathbf{q}_i\rangle &= \frac{M_i-1}{M_i} z_k^d\langle \mathbf{n}_k,\mathbf{q}_i\rangle - \frac{1}{M_i}\sum_{j\in\mathcal{N}_i} z_j^d\langle \mathbf{n}_j,\mathbf{q}_i\rangle \\ &= \frac{1}{M_i}\sum_{j\in\mathcal{N}_i}(z_k^d\langle \mathbf{n}_k,\mathbf{q}_i\rangle - z_j^d\langle \mathbf{n}_j,\mathbf{q}_i\rangle), \end{aligned} \tag{5.19}$$

then

$$\frac{1}{M_i}\sum_{j\in\mathcal{N}_i}(z_k^d\langle\mathbf{n}_k,\mathbf{q}_i\rangle - z_j^d\langle\mathbf{n}_j,\mathbf{q}_i\rangle) = (z_k^d - \tilde{z}_{a,i}^d)\langle\mathbf{n}_k,\mathbf{q}_i\rangle. \tag{5.20}$$

Define the average vector

$$\mathbf{q}_{a,i} = \frac{1}{M_i}\sum_{l\in\mathcal{H}_i}\mathbf{q}_l. \tag{5.21}$$

Hence, we write

$$\frac{1}{M_i}\sum_{j\in\mathcal{N}_i}\langle\mathbf{q}_k - \mathbf{q}_j,\mathbf{n}_i\rangle = \langle\mathbf{q}_k - \mathbf{q}_{a,i},\mathbf{n}_i\rangle, \tag{5.22}$$

and

$$\frac{1}{M_i}\sum_{j\in\mathcal{N}_i}\langle\mathbf{q}_k - \mathbf{q}_j,\mathbf{q}_i\rangle = \langle\mathbf{q}_k - \mathbf{q}_{a,i},\mathbf{q}_i\rangle. \tag{5.23}$$

Substituting (5.20), (5.18), (5.22) and (5.23) in (5.13) to obtain

$$\begin{aligned}\langle\mathbf{n}_i,\dot{\mathbf{C}}_i\mathbf{q}_i\rangle = &\sum_{k\in\mathcal{H}_i}\Big[k_1(z_k^d - \tilde{z}_{a,i}^d)\langle\mathbf{n}_k,\mathbf{n}_i\rangle + k_2\langle\mathbf{q}_k - \mathbf{q}_{a,i},\mathbf{n}_i\rangle\Big]\langle\mathbf{r}_k - \mathbf{r}_{c,i},\mathbf{q}_i\rangle\\ &+\sum_{k\in\mathcal{H}_i}\Big[k_1(z_k^d - \tilde{z}_{a,i}^d)\langle\mathbf{n}_k,\mathbf{q}_i\rangle + k_2\langle\mathbf{q}_k - \mathbf{q}_{a,i},\mathbf{q}_i\rangle\Big]\langle\mathbf{r}_k - \mathbf{r}_{c,i},\mathbf{n}_i\rangle\end{aligned} \tag{5.24}$$

Let: $z_k^d - \tilde{z}_{a,i}^d = z_k^d - z_{c,i}^d + z_{c,i}^d - \tilde{z}_{a,i}^d$. Consequently

$$\begin{aligned}\sum_{k\in\mathcal{H}_i}(z_k^d - \tilde{z}_{a,i}^d)\langle\mathbf{n}_k,\mathbf{n}_i\rangle(\mathbf{r}_k - \mathbf{r}_{c,i}) = &\sum_{k\in\mathcal{H}_i}(z_k^d - z_{c,i}^d)\langle\mathbf{n}_k,\mathbf{n}_i\rangle(\mathbf{r}_k - \mathbf{r}_{c,i})\\ &+\sum_{k\in\mathcal{H}_i}(z_{c,i}^d - \tilde{z}_{a,i}^d)\langle\mathbf{n}_k,\mathbf{n}_i\rangle(\mathbf{r}_k - \mathbf{r}_{c,i}).\end{aligned} \tag{5.25}$$

But, from (5.15), we have

$$(z_{c,i}^d - \tilde{z}_{a,i}^d)\langle \mathbf{n}_k, \mathbf{n}_i \rangle = z_{c,i}^d \langle \mathbf{n}_k, \mathbf{n}_i \rangle - \frac{1}{M_i} \sum_{l \in \mathcal{H}_i} z_l^d \langle \mathbf{n}_l, \mathbf{n}_i \rangle. \tag{5.26}$$

Then, using (5.26)

$$\begin{aligned} \sum_{k \in \mathcal{H}_i} (z_{c,i}^d - \tilde{z}_{a,i}^d)\langle \mathbf{n}_k, \mathbf{n}_i \rangle (\mathbf{r}_k - \mathbf{r}_{c,i}) &= \sum_{k \in \mathcal{H}_i} z_{c,i}^d \langle \mathbf{n}_k, \mathbf{n}_i \rangle (\mathbf{r}_k - \mathbf{r}_{c,i}) \\ &\quad - \frac{1}{M_i} \sum_{k \in \mathcal{H}_i} \sum_{l \in \mathcal{H}_i} z_l^d \langle \mathbf{n}_l, \mathbf{n}_i \rangle (\mathbf{r}_k - \mathbf{r}_{c,i}) \\ &= \sum_{k \in \mathcal{H}_i} z_{c,i}^d \langle \mathbf{n}_k, \mathbf{n}_i \rangle (\mathbf{r}_k - \mathbf{r}_{c,i}), \end{aligned} \tag{5.27}$$

where we used the fact that

$$\begin{aligned} \sum_{k \in \mathcal{H}_i} \sum_{l \in \mathcal{H}_i} z_l^d \langle \mathbf{n}_l, \mathbf{n}_i \rangle (\mathbf{r}_k - \mathbf{r}_{c,i}) &= \left(\sum_{l \in \mathcal{H}_i} z_l^d \langle \mathbf{n}_l, \mathbf{n}_i \rangle \right) \sum_{k \in \mathcal{H}_i} (\mathbf{r}_k - \mathbf{r}_{c,i}) \\ &= \left(\sum_{l \in \mathcal{H}_i} z_l^d \langle \mathbf{n}_l, \mathbf{n}_i \rangle \right) \Big(M_i \mathbf{r}_{c,i} - M_i \mathbf{r}_{c,i} \Big) \\ &= 0. \end{aligned} \tag{5.28}$$

Substituting (5.27) in (5.25)

$$\sum_{k \in \mathcal{H}_i} (z_k^d - \tilde{z}_{a,i}^d)\langle \mathbf{n}_k, \mathbf{n}_i \rangle (\mathbf{r}_k - \mathbf{r}_{c,i}) = \sum_{k \in \mathcal{H}_i} z_k^d \langle \mathbf{n}_k, \mathbf{n}_i \rangle (\mathbf{r}_k - \mathbf{r}_{c,i}) \tag{5.29}$$

We then add $0 = \sum_{k \in \mathcal{H}_i} z_c^d (\mathbf{r}_k - \mathbf{r}_{c,i})$ to (5.29) to obtain

$$\sum_{k \in \mathcal{H}_i} (z_k^d - \tilde{z}_{a,i}^d)\langle \mathbf{n}_k, \mathbf{n}_i \rangle (\mathbf{r}_k - \mathbf{r}_{c,i}) = \sum_{k \in \mathcal{H}_i} (z_k^d \langle \mathbf{n}_k, \mathbf{n}_i \rangle - z_{c,i}^d)(\mathbf{r}_k - \mathbf{r}_{c,i}) \tag{5.30}$$

Similarly,

$$\sum_{k\in\mathcal{H}_i}(z_k^d-\tilde{z}_{a,i}^d)\langle\mathbf{n}_k,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i})=\sum_{k\in\mathcal{H}_i}(z_k^d-z_{c,i}^d)\langle\mathbf{n}_k,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i})$$
$$+\sum_{k\in\mathcal{H}_i}(z_{c,i}^d-\tilde{z}_{a,i}^d)\langle\mathbf{n}_k,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i}). \tag{5.31}$$

But, from (5.15), we have

$$(z_{c,i}^d-\tilde{z}_{a,i}^d)\langle\mathbf{n}_k,\mathbf{q}_i\rangle=z_{c,i}^d\langle\mathbf{n}_k,\mathbf{q}_i\rangle-\frac{1}{M_i}\sum_{l\in\mathcal{H}_i}z_l^d\langle\mathbf{n}_l,\mathbf{q}_i\rangle. \tag{5.32}$$

Then, using (5.32)

$$\begin{aligned}\sum_{k\in\mathcal{H}_i}(z_{c,i}^d-\tilde{z}_{a,i}^d)\langle\mathbf{n}_k,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i})&=\sum_{k\in\mathcal{H}_i}z_{c,i}^d\langle\mathbf{n}_k,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i})\\&\quad-\frac{1}{M_i}\sum_{k\in\mathcal{H}_i}\sum_{l\in\mathcal{H}_i}z_l^d\langle\mathbf{n}_l,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i})\\&=\sum_{k\in\mathcal{H}_i}z_{c,i}^d\langle\mathbf{n}_k,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i}),\end{aligned} \tag{5.33}$$

where we used the fact that

$$\begin{aligned}\sum_{k\in\mathcal{H}_i}\sum_{l\in\mathcal{H}_i}z_l^d\langle\mathbf{n}_l,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i})&=\left(\sum_{l\in\mathcal{H}_i}z_l^d\langle\mathbf{n}_l,\mathbf{q}_i\rangle\right)\sum_{k\in\mathcal{H}_i}(\mathbf{r}_k-\mathbf{r}_{c,i})\\&=\left(\sum_{l\in\mathcal{H}_i}z_l^d\langle\mathbf{n}_l,\mathbf{q}_i\rangle\right)\Big(M_i\mathbf{r}_{c,i}-M_i\mathbf{r}_{c,i}\Big)\\&=0.\end{aligned} \tag{5.34}$$

Substituting (5.33) in (5.31)

$$\sum_{k\in\mathcal{H}_i}(z_k^d-\tilde{z}_{a,i}^d)\langle\mathbf{n}_k,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i})=\sum_{k\in\mathcal{H}_i}z_k^d\langle\mathbf{n}_k,\mathbf{q}_i\rangle(\mathbf{r}_k-\mathbf{r}_{c,i}) \tag{5.35}$$

We then add $0 = \sum_{k\in\mathcal{H}_i} z_c^d(\mathbf{r}_k - \mathbf{r}_{c,i})$ to (5.35) to obtain

$$\sum_{k\in\mathcal{H}_i}(z_k^d - \tilde{z}_{a,i}^d)\langle \mathbf{n}_k, \mathbf{q}_i\rangle(\mathbf{r}_k - \mathbf{r}_{c,i}) = \sum_{k\in\mathcal{H}_i}(z_k^d\langle \mathbf{n}_k, \mathbf{q}_i\rangle - z_{c,i}^d)(\mathbf{r}_k - \mathbf{r}_{c,i}) \tag{5.36}$$

Additionally, in view of (5.21)

$$\sum_{k\in\mathcal{H}_i}\langle \mathbf{q}_k - \mathbf{q}_{a,i}, \mathbf{n}_i\rangle\langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{q}_i\rangle = \sum_{k\in\mathcal{H}_i}\langle \mathbf{q}_k, \mathbf{n}_i\rangle\langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{q}_i\rangle, \tag{5.37}$$

where we used the fact that

$$\sum_{k\in\mathcal{H}_i}\langle \mathbf{q}_{a,i}, \mathbf{n}_i\rangle\langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{q}_i\rangle = \langle \mathbf{q}_{a,i}, \mathbf{n}_i\rangle\left(\sum_{k\in\mathcal{H}_i}(\mathbf{r}_k - \mathbf{r}_{c,i})^T\right)\mathbf{q}_i = 0. \tag{5.38}$$

Similarly

$$\sum_{k\in\mathcal{H}_i}\langle \mathbf{q}_k - \mathbf{q}_{a,i}, \mathbf{q}_i\rangle\langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{n}_i\rangle = \sum_{k\in\mathcal{H}_i}\langle \mathbf{q}_k, \mathbf{q}_i\rangle\langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{n}_i\rangle, \tag{5.39}$$

where we used the fact that

$$\sum_{k\in\mathcal{H}_i}\langle \mathbf{q}_{a,i}, \mathbf{q}_i\rangle\langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{n}_i\rangle = \langle \mathbf{q}_{a,i}, \mathbf{q}_i\rangle\left(\sum_{k\in\mathcal{H}_i}(\mathbf{r}_k - \mathbf{r}_{c,i})^T\right)\mathbf{n}_i = 0. \tag{5.40}$$

Substituting (5.30), (5.33), (5.37), and (5.39) in (5.24)

$$\begin{aligned}\langle \mathbf{n}_i, \dot{\mathbf{C}}_i\mathbf{q}_i\rangle = &\sum_{k\in\mathcal{H}_i}\Big[k_1(z_k^d\langle \mathbf{n}_k, \mathbf{n}_i\rangle - z_{c,i}^d) + k_2\langle \mathbf{q}_k, \mathbf{n}_i\rangle\Big]\langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{q}_i\rangle \\ &+ \sum_{k\in\mathcal{H}_i}\Big[k_1(z_k^d\langle \mathbf{n}_k, \mathbf{q}_i\rangle - z_{c,i}^d) + k_2\langle \mathbf{q}_k, \mathbf{q}_i\rangle\Big]\langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{n}_i\rangle.\end{aligned} \tag{5.41}$$

Define

$$\mathbf{w}_i = \sum_{k\in\mathcal{H}_i}\Big(z_k^d\langle \mathbf{n}_k, \mathbf{n}_i\rangle - z_{c,i}^d\Big)(\mathbf{r}_k - \mathbf{r}_{c,i}). \tag{5.42}$$

Then, substituting (5.8) in (5.41) and using (5.4) and (5.5), we obtain

$$\dot{\mathbf{n}}_i = -k_1 \frac{1}{\lambda_i^q - \lambda_i^n} \mathbf{w}_i^T \mathbf{q}_i \mathbf{q}_i - \frac{1}{\lambda_i^q - \lambda_i^n} \sigma_i \mathbf{q}_i, \tag{5.43}$$

$$\dot{\mathbf{q}}_i = k_1 \frac{1}{\lambda_i^q - \lambda_i^n} \mathbf{w}_i^T \mathbf{q}_i \mathbf{n}_i + \frac{1}{\lambda_i^q - \lambda_i^n} \sigma_i \mathbf{n}_i \tag{5.44}$$

where

$$\begin{aligned} \sigma_i = k_1 \sum_{k \in \mathcal{H}_i} \Big(z_k^d \langle \mathbf{n}_k, \mathbf{q}_i \rangle - z_{c,i} \Big) \langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{n}_i \rangle \\ + k_2 \sum_{k \in \mathcal{H}_i} \Big[\langle \mathbf{q}_k, \mathbf{n}_i \rangle \langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{q}_i \rangle + \langle \mathbf{q}_k, \mathbf{q}_i \rangle \langle \mathbf{r}_k - \mathbf{r}_{c,i}, \mathbf{n}_i \rangle \Big]. \end{aligned} \tag{5.45}$$

□

Note that, when $k_2 = 0$ and $z^d = 0$ (5.9) reduces to (4.50) derived for source seeking in Chapter 4. Additionally, $\sigma_i \to \mathbf{0}$ as $\langle \mathbf{n}_i, \mathbf{n}_j \rangle \to 1 \; \forall j \in \mathcal{N}_i$. In general, σ_i can be made arbitrary small by making k_2 small compared to k_1, i.e. slowing down the level tracking speed.

When the graph is complete, then each agent computes the same covariance matrix

$$\mathbf{C}_i = \mathbf{C} = \sum_{k=1}^{M} (\mathbf{r}_k - \mathbf{r}_c)(\mathbf{r}_k - \mathbf{r}_c)^T, \tag{5.46}$$

where all the agents see the same center $\mathbf{r}_c = \frac{1}{M} \sum_{k=1}^{M} \mathbf{r}_k$. This implies that $\mathbf{n}_i = \mathbf{n}_j = \mathbf{n}$,

and $\mathbf{q}_i = \mathbf{q}_j = \mathbf{n}$ for all i, j. Hence, $\langle \mathbf{n}_i, \mathbf{q}_j \rangle = 0$, and $\langle \mathbf{q}_i, \mathbf{q}_j \rangle = 1$ for all i, j, and thus

$$\begin{aligned}\sigma_i &= \sum_{k=1}^{M} [-k_1 z_c^d \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{n} \rangle + k_2 \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{n} \rangle] \\ &= (k_2 - k_1 z_c^d) \mathbf{n}^T \sum_{k=1}^{M} (\mathbf{r}_k - \mathbf{r}_c) \\ &= 0, \end{aligned} \tag{5.47}$$

where we used the fact that $\sum_{k=1}^{M} (\mathbf{r}_k - \mathbf{r}_c) = M\mathbf{r}_c - M\mathbf{r}_c = 0$. Additionally, using (5.8)

$$\mathbf{w} = \sum_{k=1}^{M} (z_k - z_c)(\mathbf{r}_k - \mathbf{r}_c), \tag{5.48}$$

where we used the fact that

$$z_k^d - z_c^d = z_k - z^d - \frac{1}{M} \sum_{k=1}^{M} (z_k - z^d) = z_k - z_c. \tag{5.49}$$

Observe that (5.48) is the same as (4.67) derived for the source seeking. This means that when the graph is complete, the effect of the $\mathbf{q}$ in (5.2) on the dynamics of the body frame averages out.

Using Taylor expansion, we express

$$z_k - z_c = \langle \mathbf{r}_k - \mathbf{r}_c, \nabla z_c \rangle + \nu_k, \tag{5.50}$$

where $\nabla z_c = \nabla z(\mathbf{r}_c)$ is the field gradient at the center of the swarm, and ν_k represents the higher order terms.

Then, following the same steps of deriving the equations from (4.56) till (4.65), we can obtain

$$\dot{\mathbf{n}} = -k_1 \frac{\lambda^q}{\lambda^q - \lambda^n} ||\nabla z_c|| (\mathbf{I} - \mathbf{n}\mathbf{n}^T)\mathbf{N} - \frac{k_1}{\lambda^q - \lambda^n} \sum_{k=1}^{M} \nu_k \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q} \rangle \mathbf{q}, \tag{5.51}$$

and

$$\dot{\mathbf{q}} = \frac{k_1}{\lambda^q - \lambda^n}\Big(||\nabla z_c||\lambda^q\langle \mathbf{N}, \mathbf{q}\rangle + \sum_{k=1}^{M} \nu_k \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q}\rangle\Big)\mathbf{n}. \tag{5.52}$$

where $\mathbf{N} = \frac{\nabla z_c}{||\nabla z_c||}$ is a unit length vector representing the direction of the field gradient at the center of the swarm. Interestingly, body frame dynamics under level curve tracking (5.51) and (5.52) are the same as the body frame dynamics under source seeking (4.70) and (4.64). Furthermore, recall that the term $\sum_{k=1}^{M} \nu_k \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q}\rangle$ in (5.51) and (5.52) vanishes when $\nu_k = \nu$ for all agents. i.e. the field is linear in the vicinity of the swarm's center, or when the agents are at the same level curve.

We conclude this section by the following result.

Lemma 5.4.2. *For a complete graph, for both source seeking and level curve tracking, $\lambda^n(t) \leq \lambda^q(t) = \lambda^q(t_0)$, where $\lambda^q(t_0)$ is the initial maximum variance of the spatial distribution of the agents.*

Note that this **Lemma 4.4.3** is same as **Lemma 5.4.2**. However, in proving it, we need to consider the new control law (5.2).

Proof. By the definition of PCA [54], for all unit vectors $\mathbf{u} \in \mathbf{R}^2$

$$\lambda^n = \underset{\mathbf{u}}{\operatorname{argmin}}\, \mathbf{u}^T \mathbf{C} \mathbf{u} \tag{5.53}$$

is the smallest directional variance of the spatial distribution, and

$$\lambda^q = \underset{\mathbf{u}}{\operatorname{argmax}}\, \mathbf{u}^T \mathbf{C} \mathbf{u} \tag{5.54}$$

is the largest directional variance of the spatial distribution. This implies that $\lambda^n(t) \leq \lambda^q(t)$.

What remains is to show that $\dot{\lambda}^q = 0$. Taking the time derivative of

$$\lambda^q = \mathbf{q}^T \mathbf{C} \mathbf{q} = \sum_{i=1}^{M} \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{q} \rangle^2, \tag{5.55}$$

we obtain

$$\dot{\lambda}^q = 2 \sum_{i=1}^{M} \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{q} \rangle (\langle \dot{\mathbf{r}}_i - \dot{\mathbf{r}}_c, \mathbf{q} \rangle + \langle \mathbf{r}_i - \mathbf{r}_c, \dot{\mathbf{q}} \rangle) \tag{5.56}$$

Using (5.11), for a complete graph

$$\dot{\mathbf{r}}_i - \dot{\mathbf{r}}_c = k_1 (z_i - z_a) \mathbf{n}, \tag{5.57}$$

where $z_a = \frac{1}{M} \sum_{i=1}^{M} z_i$ is the average field measurement. Hence

$$\langle \dot{\mathbf{r}}_i - \dot{\mathbf{r}}_c, \mathbf{q} \rangle = k_1 (z_i^d - z_a^d) \langle \mathbf{n}, \mathbf{q} \rangle = 0. \tag{5.58}$$

Therefore

$$\dot{\lambda}^q = 2 \sum_{i=1}^{M} \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{q} \rangle \langle \mathbf{r}_i - \mathbf{r}_c, \dot{\mathbf{q}} \rangle = 2 \langle \mathbf{C} \mathbf{q}, \dot{\mathbf{q}} \rangle = 2 \lambda^q \langle \mathbf{q}, \dot{\mathbf{q}} \rangle = 0, \tag{5.59}$$

where using (5.52), we obtain

$$\langle \mathbf{q}, \dot{\mathbf{q}} \rangle = \frac{k_1}{\lambda^q - \lambda^n} \Big(||\nabla z_c|| \lambda^q \langle \mathbf{N}, \mathbf{q} \rangle + \sum_{k=1}^{M} \nu_k \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q} \rangle \Big) \langle \mathbf{q}, \mathbf{n} \rangle = 0. \tag{5.60}$$

Similarly, taking the time derivative of

$$\lambda^n = \mathbf{n}^T \mathbf{C} \mathbf{n} = \sum_{i=1}^{M} \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n} \rangle^2, \tag{5.61}$$

we obtain

$$\dot{\lambda}^n = 2\sum_{i=1}^{M}\langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n}\rangle(\langle \dot{\mathbf{r}}_i - \dot{\mathbf{r}}_c, \mathbf{n}\rangle + \langle \mathbf{r}_i - \mathbf{r}_c, \dot{\mathbf{n}}\rangle) \tag{5.62}$$

But using (5.57)

$$\langle \dot{\mathbf{r}}_i - \dot{\mathbf{r}}_c, \mathbf{n}\rangle = k_1(z_i - z_a)\langle \mathbf{n}, \mathbf{n}\rangle = k_1(z_i - z_a), \tag{5.63}$$

and

$$\sum_{i=1}^{M}\langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n}\rangle\langle \mathbf{r}_i - \mathbf{r}_c, \dot{\mathbf{n}}\rangle = 2\langle \mathbf{C}\mathbf{n}, \dot{\mathbf{n}}\rangle = 2\lambda^n\langle \mathbf{n}, \dot{\mathbf{n}}\rangle = 0, \tag{5.64}$$

where $\langle \mathbf{n}, \dot{\mathbf{n}}\rangle = 0$ is obtained from (5.51). Substituting (5.63) and (5.63) into (5.62) yields

$$\dot{\lambda}^n = 2k_1\sum_{i=1}^{M}(z_i - z_a)\langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n}\rangle. \tag{5.65}$$

Additionally, using (5.50) and the fact that $\sum_{k=1}^{M}\langle \mathbf{r}_k - \mathbf{r}_c, \nabla z_c\rangle = 0$, we write

$$z_a = \frac{1}{M}\sum_{k=1}^{M} z_k = z_c + \frac{1}{M}\sum_{k=1}^{M}\langle \mathbf{r}_k - \mathbf{r}_c, \nabla z_c\rangle + \frac{1}{M}\sum_{k=1}^{M}\nu_k = z_c + \frac{1}{M}\sum_{k=1}^{M}\nu_k. \tag{5.66}$$

Hence

$$\dot{\lambda}^n = 2k_1\sum_{i=1}^{M}(z_i - z_c)\langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n}\rangle, \tag{5.67}$$

where we used the fact that $\sum_{i=1}^{M}\sum_{k=1}^{M}\nu_k\langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n}\rangle = \sum_{k=1}^{M}\nu_k\sum_{i=1}^{M}\langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n}\rangle = 0$.

Finally, using (5.50) for $(z_i - z_c)$

$$\begin{aligned}\dot{\lambda}^n &= 2k_1||\nabla z_c||\mathbf{N}^T\sum_{i=1}^{M}(\mathbf{r}_i-\mathbf{r}_c)(\mathbf{r}_i-\mathbf{r}_c)^T\mathbf{n} + 2k_1\sum_{i=1}^{M}\nu_i\langle\mathbf{r}_i-\mathbf{r}_c,\mathbf{n}\rangle \\ &= 2k_1\lambda^n||\nabla z_c||\langle\mathbf{N},\mathbf{n}\rangle + 2k_1\sum_{i=1}^{M}\nu_i\langle\mathbf{r}_i-\mathbf{r}_c,\mathbf{n}\rangle. \qquad (5.68)\end{aligned}$$

□

Note that, from (5.68), λ^n will increase or decrease, hence the shape will stretch or shrink along the $\mathbf{n}$ direction, depending on the signs of $\langle\mathbf{N},\mathbf{n}\rangle$ and $\sum_{i=1}^{M}\nu_i\langle\mathbf{r}_i-\mathbf{r}_c,\mathbf{n}\rangle$. However, if it increases, it will do so only up to $\lambda^n = \lambda^q$. At this point, the PCA flow will interchange $\mathbf{n}$ and $\mathbf{q}$ and hence the swarm performs a turn of at most 90°. Importantly, **Lemma 5.4.2** implies that the connectivity of the graph is maintained implicitly.

5.5 Convergence Analysis

We want to prove that using the PCA flow (4.3), the control (5.2) steers the agents towards the desired level curve when $k_2 \neq 0$.

Recall that the PCA flow (4.3) runs in the time scale τ, while the swarming control law (5.2) runs in the time scale t. That is, for each time instance t, each agent runs (4.3) for some time τ. To overcome this difficulty, we formulate a singular perturbation problem as follows. Let the relationships between the swarming time t, and the PCA learning time τ be of the form $\frac{dt}{d\tau} = \epsilon$, where $\epsilon \in (0,1)$. This implies that $\tau = \frac{t-t_0}{\epsilon}$, where $\tau_0 = 0$. In other words, τ is stretched as $\epsilon \to 0$, and shrunk as $\epsilon \to 1$. Using this relationship, the PCA learning and swarming dynamics in the singular perturbation framework are

$$\frac{d\mathbf{r}_i}{dt} = k_1(z_i(t)-z_d)\mathbf{n}_i(t) + k_2\mathbf{q}_i(t), \quad \forall i \qquad (5.69)$$

$$\epsilon\frac{d\mathbf{q}_i}{dt} = (\mathbf{I}-\mathbf{q}_i(\tau)\mathbf{q}_i^T(\tau))\mathbf{C}_i(t)\mathbf{q}_i(\tau), \quad \forall i, \qquad (5.70)$$

where the swarming dynamics (5.69) is viewed as a slow system, and the PCA learning dynamics (5.70) is viewed as a fast system. In this framework, we first let $\epsilon = 0$ in (5.70) to obtain decoupled reduced and boundary systems. We then analyze the stability of the origins of these decoupled systems. Finally, we derive $\epsilon^* \in (0, 1)$ such that for all $\epsilon \leq \epsilon^*$, the stability results of the reduced and boundary systems hold for the original slow (5.69) and fast (5.70) systems. Practically, ϵ^* reveals how rapid the fast systems must be in order for the system to converge to the desired equilibrium. In what follows, we first analyze the source seeking and level curve tracking in complete graphs and then generalize the source seeking to incomplete graphs.

Define

$$\zeta = z_c - z^d. \tag{5.71}$$

This implies that $\zeta = 0$ if and only if $z_c = z^d$. Taking the time derivative

$$\dot{\zeta} = \dot{z}_c = \langle \nabla z_c, \dot{\mathbf{r}}_c \rangle. \tag{5.72}$$

But, using (5.69) and (5.50), we have

$$\dot{\mathbf{r}}_c = \frac{1}{M} \sum_k [k_1 (z_k - z^d)\mathbf{n} + k_2 \mathbf{q}] = k_1 (\zeta + \frac{1}{M} \sum_k \nu_k)\mathbf{n} + k_2 \mathbf{q} \tag{5.73}$$

Then we obtain the reduced system for the level curve tracking

$$\begin{aligned} \dot{\zeta} &= k_1 ||\nabla z_c|| (\theta - 1)(\zeta + \frac{1}{M} \sum_{k=1}^{M} \nu_k) + k_2 ||\nabla z_c|| \langle \mathbf{N}, \mathbf{q} \rangle, \\ \dot{\theta} &= \frac{k_1 \lambda^q ||\nabla z_c||}{\lambda^q - \lambda^n} \theta(\theta - 2) + \frac{k_1}{\lambda^q - \lambda^n} \Delta + \delta, \end{aligned} \tag{5.74}$$

where where $\Delta = -\langle \mathbf{N}, \mathbf{q} \rangle \sum_{k=1}^{M} \nu_k \langle \mathbf{r}_k - \mathbf{r}_c, \mathbf{q} \rangle$ is due to the nonlinear components of the field, and $\delta = \langle \mathbf{n}, \dot{\mathbf{N}} \rangle$ is viewed as an unknown input field disturbance. Note that $\dot{\theta}$ is the

same for both source seeking and level curve tracking. This is because $\dot{\mathbf{n}}$ and $\dot{\mathbf{q}}$ derived in (5.51) and (5.52) are derived for both source seeking and level curve tracking. The following result summarizes the stability of the reduced system.

Lemma 5.5.1. *Consider the reduced system (5.74) and suppose* $0 \leq \theta(0) \leq 1$. *Then the origin of the unforced system* $f(\zeta, \theta, 0)$ *is ultimately bounded. Furthermore, suppose that* $\Big(k_1||\nabla z_c||\lambda^q|\nu| + k_1|\Delta| + (\lambda^q - \lambda^n)|\delta|\Big)/(\epsilon_1 k_1||\nabla z_c||\lambda^q) < \zeta(0)^2$, *then the origin of forced system* $f(\zeta, \theta, \delta)$ *is input-to-state stable.*

Proof. Consider the domain $D_1 = \{[\zeta, \theta]^T | \theta \in [0, 1)\}$. Let $V_1 : D_1 \to \mathbf{R}$ be a Lyapunov candidate function defined by

$$V_1 = \frac{1}{2}\zeta^2 + \Big(\frac{\lambda^q - \lambda^n}{\lambda^q}\Big)\Big(\frac{\theta}{1-\theta}\Big), \tag{5.75}$$

where $V_1 = 0$ if and only if $[\zeta, \theta]^T = [0, 0]^T$. Additionally, $V_1 \to \infty$ as $\theta \to 1$.
For the unforced system $f(\mathbf{x}, 0)$, we set $\delta = 0$ which implies that $\nu_k = 0$. Hence, taking the derivative of V_1 and using (5.68) and (5.74), we obtain

$$\dot{V}_1 = -k_1||\nabla z_c||\Big((1-\theta)\zeta^2 + \theta[\frac{2-\theta}{(1-\theta)^2} - 2\frac{\lambda^n}{\lambda^q}]\Big) + k_2||\nabla z_c||\zeta\langle \mathbf{N}, \mathbf{q}\rangle.$$

First note that $\dot{V}_1 = 0$ if and only if $[\zeta, \theta] = [0, 0]$. Next, when $k_2 = 0$, then $\dot{V}_1 \leq 0$ which implies that the unforced system will converge to a position where $\zeta = 0$, i.e. at the desired level curve. When $k_2 \neq 0$, we have $\dot{V}_1 \to -k_1||\nabla z_c||\zeta^2$ as $\theta \to 0$, and $\dot{V}_1 \to -\infty$ as $\theta \to 1$. Furthermore, let $\epsilon_1 \in (0, 1)$, then for $|\zeta| \geq \frac{k_2}{k_1\epsilon_1}$, we obtain $\dot{V}_1 \leq -k_1(1-\epsilon_1)||\nabla z_c||(1-\theta)\zeta^2 \leq 0$. This means that, according to **Theorem 4.18** in [68], the unforced system $f(\zeta, \theta, 0)$ is ultimately bounded, and hence the trajectory of the center of the swarm will converge to a strip defined by $|\zeta| < \frac{k_2}{k_1\epsilon_1}$. The width of the strip can be made arbitrary small by making k_2 small enough compared to k_1. Additionally, since $\dot{V}_1 \to -\infty$ as $\theta \to 1$ and $V_1 \to \infty$ whenever $\theta \to 1$, then $\mathbf{D}_1$ is a forward invariant set.

For the forced system $f(\zeta, \theta, \delta)$, let $W_1(\zeta, \theta)$ be a continuous positive definite function as defined in (4.96) in which z_c is replaced by ζ. Consequently, we obtain

$$\dot{V}_1 \leq -(1-\epsilon_1)W_1, \quad \forall |\zeta| > \max\{\rho(|\delta|), \frac{k_2}{k_1\epsilon_1}\}, \tag{5.76}$$

where $\rho(|\delta|)$ is as defined in (4.98). Let $\alpha_1(||[z_c, \theta]^T||) = \alpha_2(||[z_c, \theta]^T||) = \frac{1}{2}|\zeta|^2 + \frac{\lambda^q - \lambda^n}{\lambda^q}\frac{|\theta|}{1-|\theta|}$ which are class $\mathcal{K}_\infty$ functions that satisfy: $\alpha_1(||[z_c, \theta]^T||) \leq V_1([z_c, \theta]^T) \leq \alpha_2(||[z_c, \theta]^T||)$. Therefore, according to *Theorem 4.19* in [68], the origin of the forced system $f(\zeta, \theta, \delta)$ is input-to-state stable. □

Since for the level curve tracking, we are still using the same PCA flow (4.88), we have the same boundary system obtained in Section 4.5.1. Therefore, **Lemma 4.5.2** holds when $k_2 \neq 0$ and $z^d \neq 0$.

Finally, define the coupled system $[\dot{\zeta}, \dot{\theta}, \dot{\psi}] = h(\epsilon, \zeta, \theta, \psi, \delta)$ where $\dot{\zeta}$ and $\dot{\theta}$ are given by (5.74), and $\dot{\psi}$ is given by (4.103). The following theorem establishes a sufficient range for ϵ that generalizes **Lemma 5.5.1** and **Lemma 4.5.2** of the decoupled system $h(0, \zeta, \theta, \psi, \delta) = f(\zeta, \theta, \psi, \delta)$ to the coupled system $h(\epsilon, \zeta, \theta, \psi, \delta)$.

Proposition 5.5.1. *Consider the coupled system given by (4.87) and (4.88) where $k_2 \neq 0$ and $z^d \neq 0$. Suppose that $\epsilon \leq \epsilon^*$ where*

$$\epsilon^* = \frac{\epsilon_2 \lambda^q(\lambda^q - \lambda^n)}{k_1\epsilon_z\lambda^q + k_1\sum_k \nu_k||\mathbf{r}_k - \mathbf{r}_c||}. \tag{5.77}$$

If $0 \leq \theta(0) \leq 1$ and $0 \leq \psi(0) \leq 1$, then the origin of the unforced system $h(\epsilon, \zeta, \theta, \psi, 0)$ is ultimately bounded. Furthermore, if $0 \leq \theta(0) \leq 1$, $0 \leq \psi(0) \leq 1$, and $\Big(k_1||\nabla z_c||\lambda^q|\nu| + k_1|\Delta| + (\lambda^q - \lambda^n)|\delta|\Big)/(\epsilon_1 k_1||\nabla z_c||\lambda^q) < \zeta(0)^2$, then the origin of forced system $h(\epsilon, \zeta, \theta, \psi, \delta)$ is input-to-state stable.

Proof. Applying the same procedure in Section 4.5.1, this results follows from **Lemma 4.5.2** and **Lemma 5.5.1**. □

Note that, as expected, since we are using the same PCA flow (5.70), we obtained the same bound ϵ^* for both the source seeking and curve tracking algorithms.

5.6 Simulation and Experimental Results

In this section, we validate the proposed model through computer simulation and physical experiments. We used two robotic platforms: the surface Georgia Tech Robotarium unicycle robots [22], and the air Georgia Tech Miniature Autonomous Blimps [23]. In what follows, we first present the simulation results and then the experimental results. In all simulations and experiment, we set $\epsilon = 0.01$ in the (5.70). This means that we run the PCA flow (4.3) for a time $\tau = \frac{dt}{\epsilon}$ where $dt = 0.01$ is the step time used to update (5.2).

5.6.1 Simulation Results

We simulate the strategy for different scenarios. In all scenarios, we set $z_d = 2$ and $\epsilon = 0.01$. In Fig. 5.2 and Fig. 5.3, a 2-agent system is simulated in a convex field for different k_2 values. As predicted by **Proposition 5.5.1**, agents in Fig. 5.3 track the desired level curve with more accuracy than that of Fig. 5.2, however, they move slowly and hence for the same amount of time they passed shorter segment of the level curve. The 2-agent system

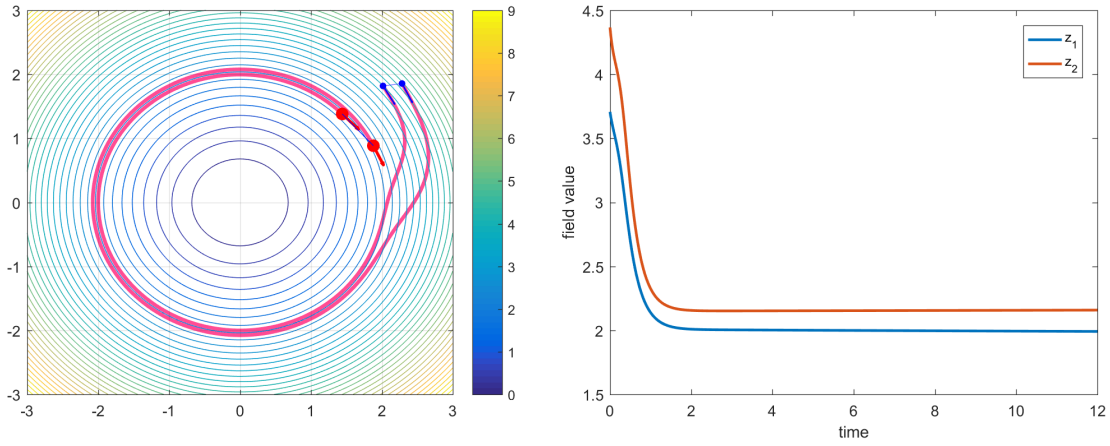


Figure 5.2: 2-agent system level curve tracking with $k_1 = 2$, $k_2 = 1$, $t_f = 12$ seconds

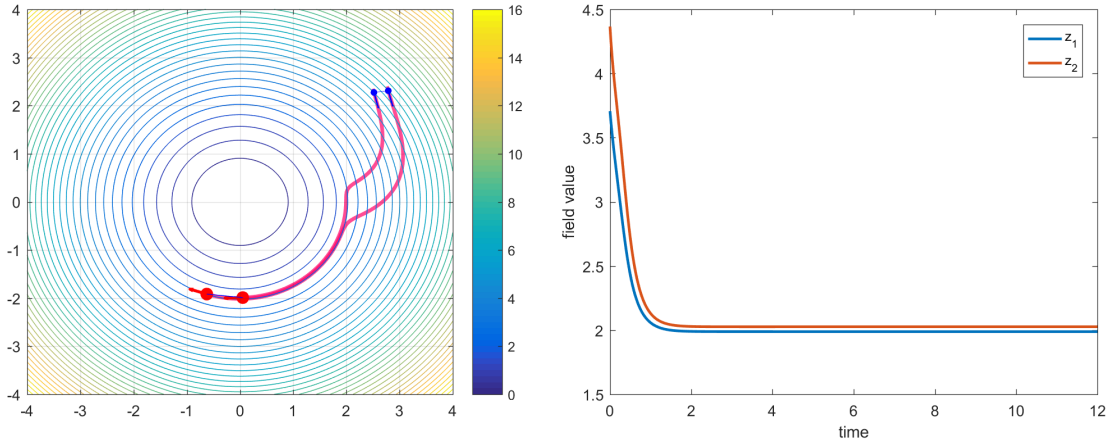


Figure 5.3: 2-agent system level curve tracking with $k_1 = 2$, $k_2 = 0.3$, $t_f = 12$ seconds

in Fig. 5.4 successfully tracks the desired level curve, but with varying deviation from the desired value that is attributed to the non-convexity of the field.

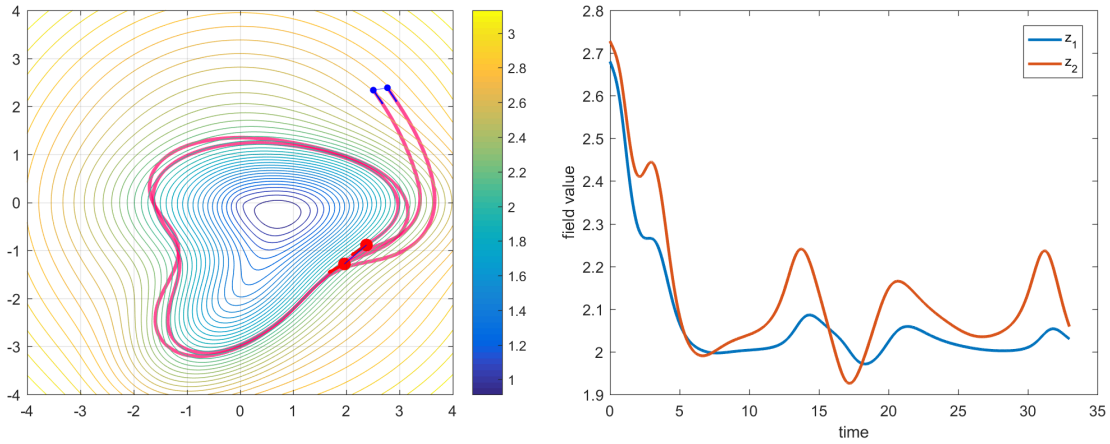


Figure 5.4: 2-agent system level curve tracking with $k_1 = 2$, $k_2 = 0.5$, $t_f = 33$ seconds

Alternatively, in Fig. 5.5, a 7-agent system with a line graph is simulated in a convex field (left) and a non-convex field (right). Since the agent are closer to each other, we observe that the overshoot is much less compared to the 2-agent system in Fig. **??**.

In Fig. 5.6, a 10-agent system with an arbitrary static and connected graph is simulated. Since the agents are more spatially distributed, they steer to the desired level curve relatively faster than the 2-agent systems in Fig. 5.2 and Fig. 5.3. Alternatively, In Fig. 5.7, each agent chooses the closest three agents as its neighbors at each instant of time. This

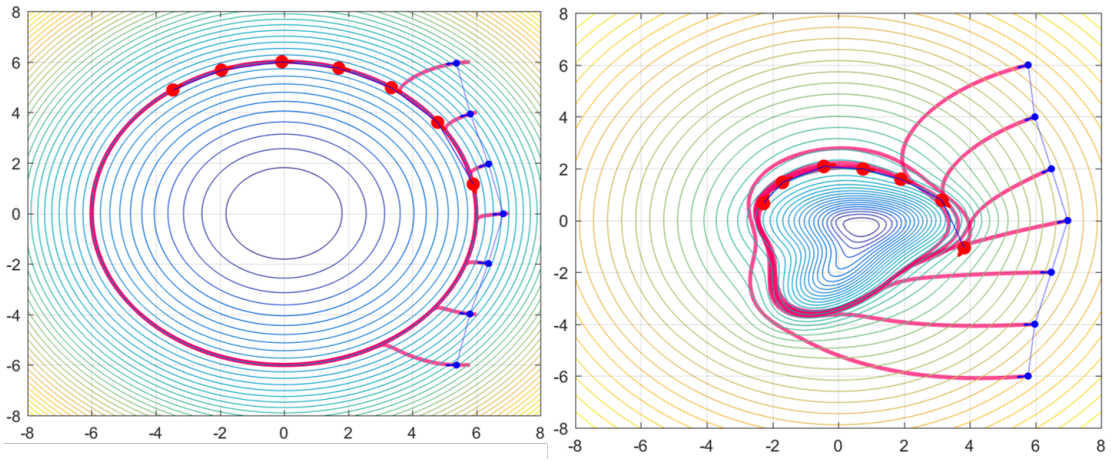


Figure 5.5: a 7-agent system with a line graph in a convex field (left) and a non-convex field (right).

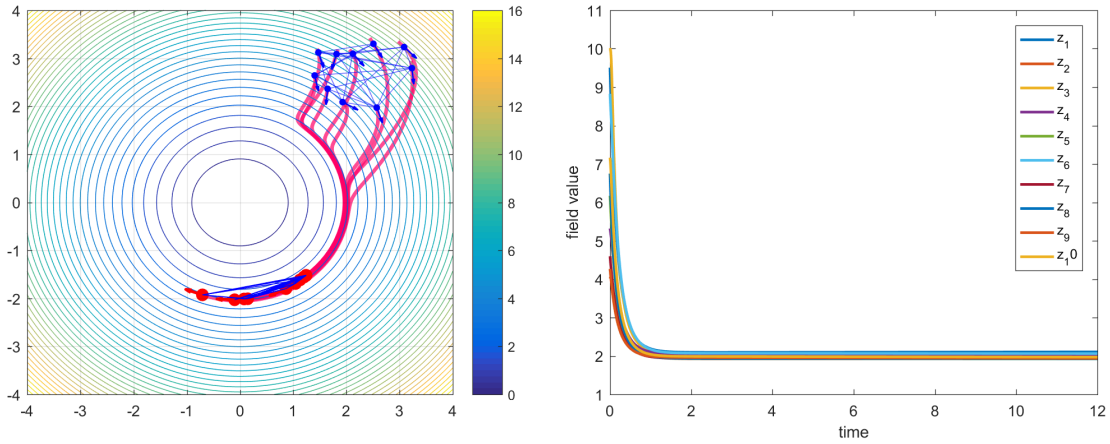


Figure 5.6: Static graph of A 10-agent system level curve tracking with, $k_1 = 2$, $k_2 = 0.3$, $t_f = 12$ seconds

leads to a directed and dynamic graph.

5.6.2 Experimental Results

We implemented the level curve tracking algorithm with convex and non-convex fields using the Robotarium robots and the results are shown in Fig. 5.8 and Fig. 5.9, respectively. In these two figures, we projected the resulting robots' trajectories which clearly match the simulated ones in Fig. 5.2 and Fig. 5.4, respectively.

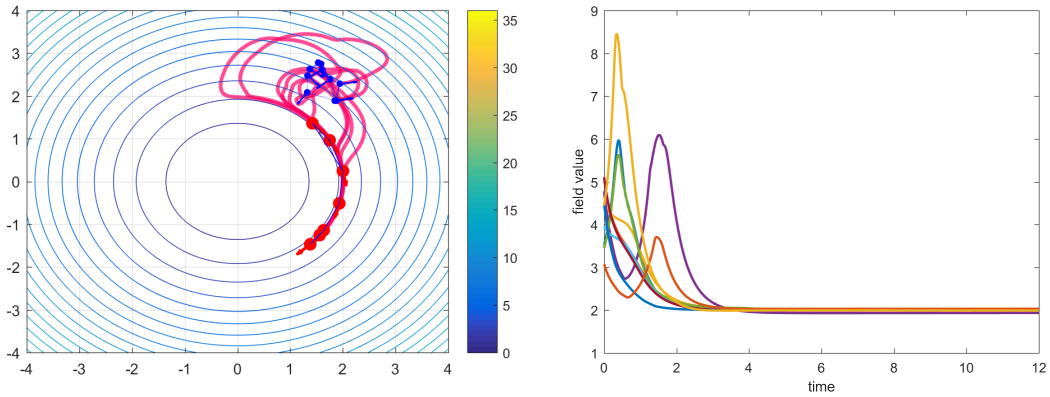


Figure 5.7: Dynamic graph of a 10-agent system level curve tracking with, $k_1 = 1$, $k_2 = 0.2$, $t_f = 12$ seconds

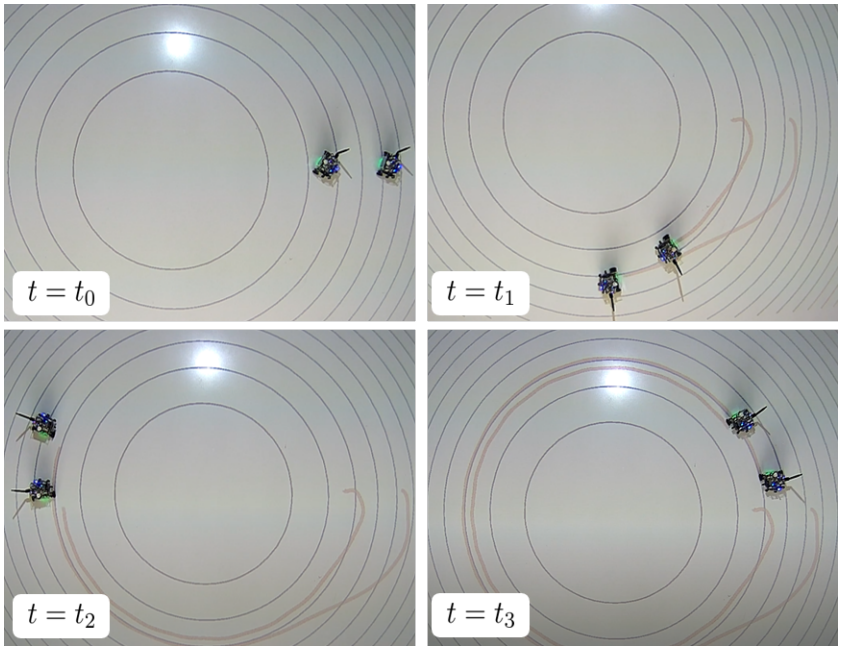


Figure 5.8: Two robots in a convex field.

5.7 Conclusion

In this chapter, we designed a distributed control law for level curve tracking based on the Multi-Layer control model proposed on Chapter 4. The control law enables swarms of various sizes and graph structures to perform collective level curve tracking of scalar fields without the need to explicitly estimate the field gradient or explicitly share measurements among the agents. We obtained stability results in a singular perturbation framework justi-

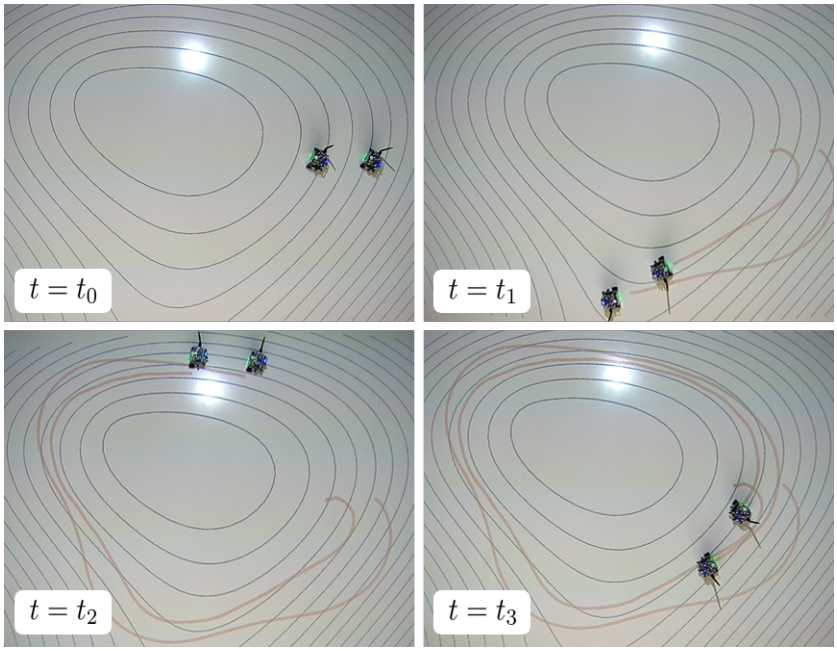


Figure 5.9: Two robots in a non convex field.

fying the robustness and convergence of the algorithms. The simulation and experimental results suggest the efficiency and generality of the proposed model.

CHAPTER 6
FROM SOURCE SEEKING AND LEVEL CURVE TRACKING TO INFORMATION PROPAGATION AND MULTI-TASKING

6.1 INTRODUCTION

Biological research has established evidence that critical decisions such as performing a turn in a flock of birds or avoiding a predator in a school of fish are initiated by individuals at the boundary of the swarm and then propagate to the entire collective [2, 3, 1]. The propagating signal might take the form of a change of motion behavior of an agent to alert the swarm about a predator or guide the swarm to a possible source of food, or to a shelter such as a dark area, as in the case of shiner fish schools [16]. The question is how this information propagation occurs almost instantly and presumably without explicit communication between the individuals even when they are significantly far away from where the signal starts. Even more surprising is how these external signals, for example, predator warnings, occur and propagate while the individuals are collectively performing other complex tasks such as foraging or synchronization.

In this chapter, we propose a Multi-Layer control model composed of an interplay of decentralized algorithms for perception and swarming. Through this novel model, we demonstrate implicit information propagation and multi-tasking in swarms using only local interactions and without explicit communication. In the perception layer, each agent not only applies a Principal Component Analysis (PCA) on the relative positions of its neighbors, as in Chapter 4, but also applies PCA on relative headings of its neighbors to synchronize motion. These principal values are then used in the swarming layer where a distributed control law is designed to balance between achieving a collective task and at the same time allowing critical emerging signals to propagate to the entire swarm.

In [1], its authors explore how a school of shiner fish gain information from the changes in position and heading of their neighbors when attacked by a robotic predator fish. In [2, 3], their authors studied how collective turning emerges from agents' fluctuations in natural flocks of starlings. They experimentally validated through real data that turning starts from the agents who are located at the elongated tips of the flock, and then propagates across the remaining agents. In [44], by analyzing real trajectories of Killifish those authors showed that fish coordinate their position and speed in a way that enriches social communication. In particular, oscillations in speed lead to spatial geometrical configurations such as a V formation and a diagonal formation. All of these works support the idea that the spatial mean and variance seen by each agent are valuable elements for information propagation in swarms exhibiting collective behaviors.

Our proposed model treats sudden changes in the motion of the individuals as a valuable source of information. These changes often are viewed by existing models as undesired disturbances that are suppressed by firm consensus or formation control laws for the sake of rigid synchronization or stability. In the swarming layer, we design the motion controllers such that each individual focuses more on synchronization with its neighbors when internal or external stimuli are low. On the other hand, attention is reversed to respond to the locally observed changes when internal or external stimuli are high. Most of the existing results that deal with the propagation of turn waves in natural swarms use the Vicsek model [18]. However, authors in [19, 20] developed a model based on statistical physics that they argue better explains wave propagation than the Vicsec model, in which the turning signals get attenuated and might not reach the entire swarm. Instead of using the Vicsec model or the common consensus-on-a-sphere to synchronize [8, 21, 51], we develop a synchronizing algorithm based on PCA of locally observed headings. The consensus-on-a-sphere always converges to the average heading. However, the PCA on headings tends to pay more attention to the outliers. This results in an agile and flexible motion behavior where individuals promptly respond to stimuli while achieving a collective task. Remarkably, information

propagation, as well as individual responses, occur implicitly without explicitly enforcing rigid formations.

The main contribution of this chapter is a self-tuned distributed control law that incorporates a PCA learning algorithm on relative positions to respond to local changes, and a PCA learning algorithm on headings to synchronize motion with the swarm. The control law is scalable to swarms of various sizes and graph structures. For a complete graph, we show that variations on individual speed signals are implicitly propagated across the swarm and enforce a change in direction on the whole swarm almost instantly. Additionally, we prove that the spatial variances of the shape of the swarm are bounded which implicitly ensure connectivity of the graph. Finally, we provide various simulation results demonstrating the effectiveness of the model for swarms with complete and incomplete graphs performing collective synchronization and source seeking while avoiding a predator [17].

The first difficulty this chapter sets out to solve is how to detect critical information in motion behavior using only local measurements. The PCA algorithms in the perception layer allow each individual to extract valuable knowledge from the raw data of relative positions and headings that effectively capture variations in the behavior of its neighbors. The second difficulty is how to respond in a way that allows the information to propagate while at the same time performing a collective task. We solve this by allocating some part of the control effort to respond to the local changes while allocating the remaining part to synchronize with the other individuals. Finally, the perception algorithms run in a time-scale that is different and faster than the swarming algorithms. We accommodate this difficulty by employing a singular perturbation framework to derive the necessary dynamics as well as providing stability results.

Designing analytical models to understand information propagation will not only reveal natural mysteries but additionally will help to propose multi-tasking control algorithms for robotic swarms that require only very limited or no explicit communication. In particular, this is highly related to the problem of designing tactics for a swarm of drones to avoid or

chase a malicious agent [24, 25]. The proposed Multi-Layer model might potentially be used to design various swarm algorithms, especially those incorporating individual differences between agents.

The rest of the chapter is organized as follows. The problem is formulated in Section 6.2. The Multi-Layer model including the algorithms for perception and swarming are presented in Section 6.3. Then, dynamics and stability analysis and simulation results are given in Section 6.5 and Section 6.6 respectively. Finally concluding remarks and suggestions for future work are provided in Section 6.7.

6.2 Problem Formulation

6.2.1 Preliminaries

Consider a swarm of M agents. Let the interaction among the agents be described by an undirected graph, $\mathcal{G} \subseteq \mathcal{V} \times \mathcal{E}$ where $\mathcal{V}$ is the set of all agents, and $\mathcal{E}$ is the set of all edges. An undirected edge $(i, j) \in \mathcal{E}$ exists if both agents can sense the relative positions of each other. A graph is connected if for each $i, j \in \mathcal{V}$, there exists a sequence of edges connecting the i^{th} and j^{th} agents. If each agent shares an edge with all other agents, then the graph is complete. A Delaunay graph is formed by the edges connecting a planar triangulation of all vertices such that any circumcircle of a triangle includes only the vertices of that triangle. The neighbor set of i is defined by $\mathcal{N}_i = \{j | (i, j) \in \mathcal{E}\}$. Additionally, if for each agent $\mathcal{N}_i$ is fixed, the graph is static, otherwise, it is dynamic. We consider the following assumption about the graph:

Assumption 6.2.1. *$\mathcal{G}$ is undirected and connected and is either a complete graph or a Delaunay graph.*

Although a complete graph is suitable only for swarms of small size, we will use it to provide motivations and insights for the proposed model as well as some stability results. On the other hand, biological research reveals that Delaunay graphs are more suitable than

metric and topological graphs to describe information propagation when individuals have only visual sensors [73].

Let $\mathbf{r}_i \in \mathbb{R}^2$ be the position of the i^{th} agent. Assume that:

Assumption 6.2.2. *each agent is able to measure the relative positions* $\mathbf{r}_j, \forall j \in \mathcal{N}_i$.

In practice, robots can be equipped with sensors such as a compass and a range sensor to measure the relative positions, which is less challenging than requiring the global positions [5]. Additionally, let $\mathbf{h}_j$ be a unit length vector representing the velocity direction of agent j as seen by agent i. We then assume

Assumption 6.2.3. *each agent is able to measure the relative headings* $\mathbf{h}_j, \forall j \in \mathcal{N}_i$.

Note that for this assumption we only require the direction of the velocity, which in fish or birds represents the orientation of the animal. Additionally, using advanced visualization techniques, agents can estimate the headings of their neighbors [65].

6.2.2 Problem Statement

Consider the collective behaviors shown in Figs. 6.1, 6.2 and 6.3. In Fig. 6.1 the swarm is initially exhibiting an arbitrary synchronized motion and then one agent initiated a turning signal in response to an approaching predator. Alternatively, in Fig. 6.2, the agent in the front of the swarm initiated a turning signal for an unspecified reason which then propagated to the entire swarm. Finally, in Fig. 6.3, the swarm performing source seeking is forced to change its direction as some agents detect a predator.

The problem of this chapter is to design a distributed control law $\mathbf{u}_i$ such that when agents move according to:

$$\dot{\mathbf{r}}_i = \mathbf{u}_i, \quad i = 1, \cdots, M, \tag{6.1}$$

the swarm pursues a collective task while important decisions propagates across it as depicted in the scenarios of Figs. 6.1, 6.2 and 6.3.

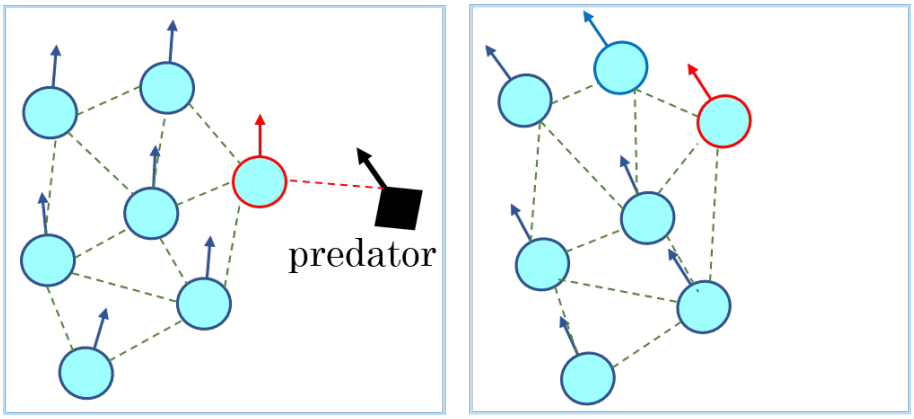


Figure 6.1: A predator in the left figure is detected by one agent (outlined in red) which initiates a turn. The turn then propagates to the entire swarm pushing it to move away from the predator.

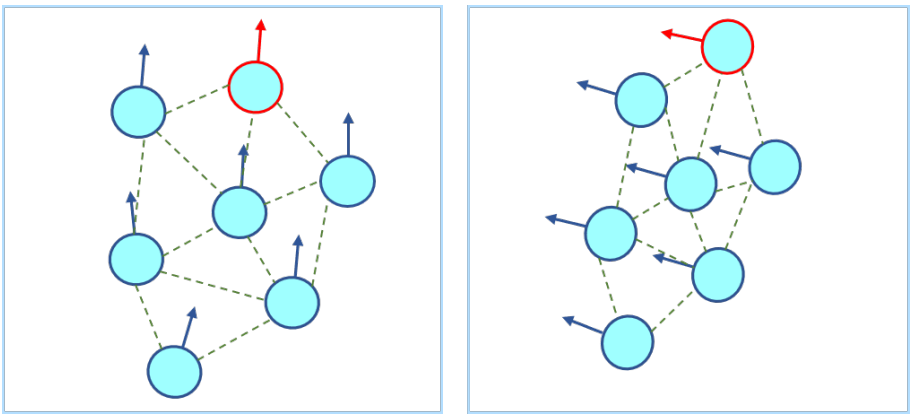


Figure 6.2: The agent at the front initiates a turn that propagates to the entire swarm.

6.3 The Multi-Layer Model

The proposed Multi-Layer model is shown in Fig. 6.4. The upper two perception layers are used to learn directions to be used in modulating the velocity in the bottom swarming layer. In what follows we will explain first the PCA Perception Algorithms and then the Swarming Algorithms.

6.3.1 The PCA Perception Algorithms

Principal Component Analysis (PCA) is a statistical method that computes directions of maximum (minimum) variation of a data set [54]. Given a covariance matrix of a data set, its eigenvector corresponding to the maximum (minimum) eigenvalue represents the

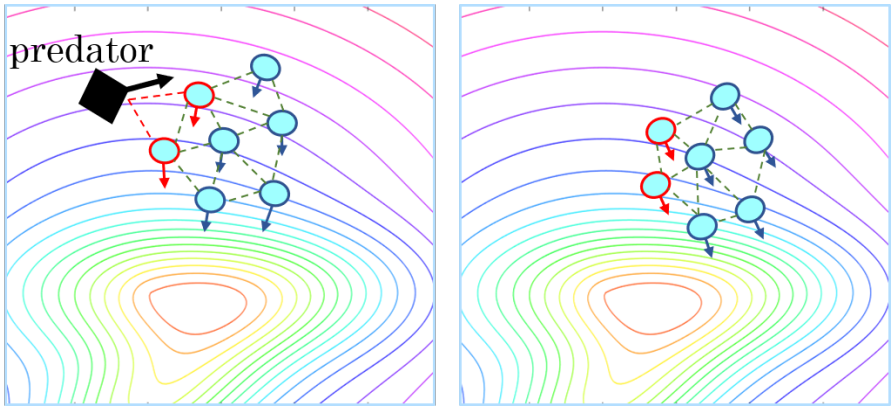


Figure 6.3: While the swarm is performing source seeking, agents on the perimeter detect a predator and their reaction causes the entire swarm to change direction.

direction of the maximum (minimum) variance of the data, with the eigenvalue giving the variance of the data along that direction.

In the first Perception Layer of Fig. 6.4, agents apply PCA on relative positions of their neighbors, while in the second Perception Layer they apply PCA on relative headings of their neighbors. For each agent, consider the set defined by $\mathcal{F}_i = \mathcal{N}_i \bigcup \{i\}$. Then, the position covariance matrix $\mathbf{C}_i(t) \in \mathbb{R}^{2\times 2}$ of each agent is defined by

$$\mathbf{C}_i(t) = \sum_{j\in\mathcal{F}_i} (\mathbf{r}_j(t) - \mathbf{r}_{c_i}(t))(\mathbf{r}_j(t) - \mathbf{r}_{c_i}(t))^T, \tag{6.2}$$

where $\mathbf{r}_{c_i} = \frac{1}{|\mathcal{F}_i|}\sum_{j\in\mathcal{H}_i} \mathbf{r}_j$ is the center of agents seen by agent i. Each agent obtains its principal directions by finding the eigenvectors of (6.2). We illustrate the concept in Fig. 6.5 wherein agent i has three immediate neighbors. The spatial largest and smallest principal axes are denoted by PC1 and PC2, respectively. Since each agent may have different neighbors, each one obtains different principal axes. To obtain the principal components of (6.2), we utilize an unsupervised learning algorithm that is described by a dynamical model called the one-unit Oja PCA Flow model [14, 15]. Let $\mathbf{q}_i(t)$ and $\mathbf{n}_i(t)$ be orthonormal vectors in $\mathbb{R}^2$ that represent the principal components of the covariance matrix $\mathbf{C}_i(t)$, corresponding to the largest and smallest eigenvalues, λ_i^q and λ_i^n, respectively. Then the Oja PCA Flow

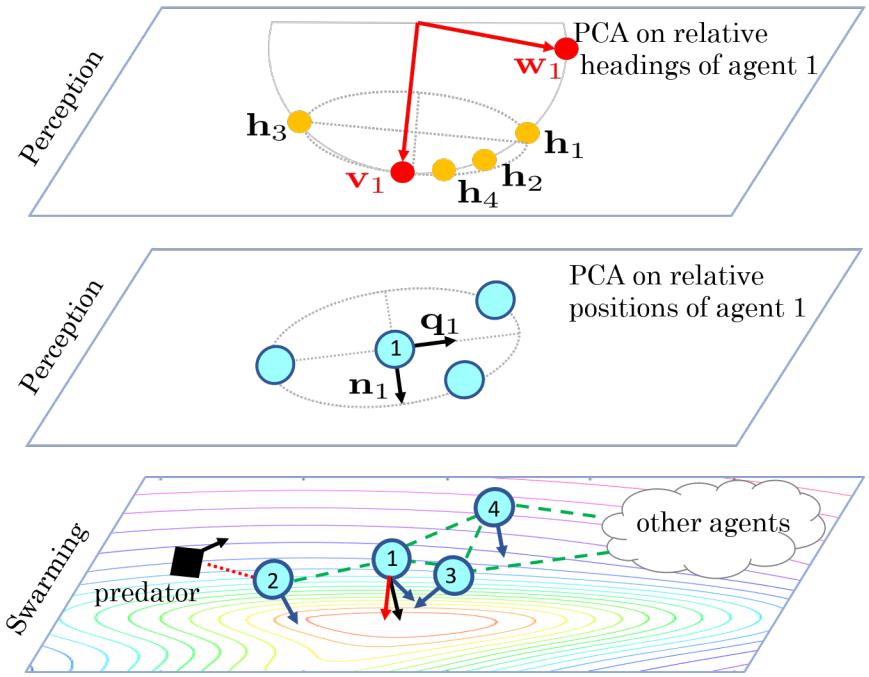


Figure 6.4: In the Swarming Layer agents modulate their motion on directions obtained by the two Perception Layers.

model is

$$\begin{aligned}\dot{\mathbf{q}}_i(\tau) &= (\mathbf{I} - \mathbf{q}_i(\tau)\mathbf{q}_i^T(\tau))\mathbf{C}_i(t)\mathbf{q}_i(\tau) \\ \mathbf{n}_i(\tau) &= \mathbf{R}\mathbf{q}_i(\tau),\end{aligned} \tag{6.3}$$

where $\mathbf{R}$ is a 90° counterclockwise rotation matrix. Each agent forms its body frame as $(\mathbf{q}_i(t), \mathbf{n}_i(t))$. Importantly, the PCA model (6.3) is scalable to graphs with arbitrary number of agents and structures, as long as each agent has at least one neighbor. Observe that we use the argument τ instead of t to emphasize that for any given $\mathbf{C}_i(t)$ at time instant t, agent i runs (6.3) at a different time scale τ. It is known that the trajectories of (6.3) asymptotically converge to the principal components of $\mathbf{C}_i(t)$ almost everywhere [13, 56]. That is, $\mathbf{q}(\tau) \to \mathbf{q}(\mathbf{t})$ as $\tau \to \infty$ asymptotically.

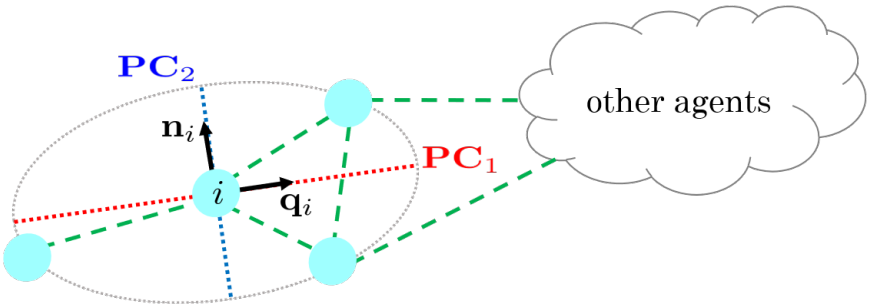


Figure 6.5: The green dashed lines are the edges. Agent i sees three neighbors, from which it forms the largest and smallest principal components PC1 and PC2, respectively.

Similarly, define the heading covariance matrix $\mathbf{H}_i(t) \in \mathbb{R}^{2\times 2}$ of each agent to be

$$\mathbf{H}_i(t) = \sum_{j\in\mathcal{F}_i} (\mathbf{h}_j(t) - \mathbf{h}_{c_i}(t))(\mathbf{h}_j(t) - \mathbf{h}_{c_i}(t))^T, \tag{6.4}$$

where $\mathbf{h}_j$ is a unit length vector representing the velocity direction of agent j as seen by agent i, and $\mathbf{h}_{c_i} = \frac{1}{|\mathcal{H}_i|}\sum_{j\in\mathcal{H}_i} \mathbf{h}_j$ is the average velocity direction as seen by agent i. Let $\mathbf{w}_i(t)$ and $\mathbf{v}_i(t)$ be orthonormal vectors in $\mathbb{R}^2$ that represent the principal components of the covariance matrix $\mathbf{H}_i(t)$, corresponding to the largest and smallest eigenvalues, λ_i^w and λ_i^v, respectively. Then using the Oja PCA Flow model

$$\begin{aligned} \dot{\mathbf{w}}_i(\tau) &= (\mathbf{I} - \mathbf{w}_i(\tau)\mathbf{w}_i^T(\tau))\mathbf{H}_i(t)\mathbf{w}_i(\tau) \\ \mathbf{v}_i(\tau) &= \mathbf{R}\mathbf{w}_i(\tau), \end{aligned} \tag{6.5}$$

where $\mathbf{R}$ is a 90° counterclockwise rotation matrix. Note that the heading vectors are of unit length which we can represent as points on the circumstance of a unit circle as illustrated in Fig. 6.6. In this figure agent i and its three immediate neighbors are represented by yellow discs. Then $\mathbf{w}_i$ and $\mathbf{v}_i$ are unit length vectors along the spatial largest and smallest principal axis, PC1 and PC2, respectively.

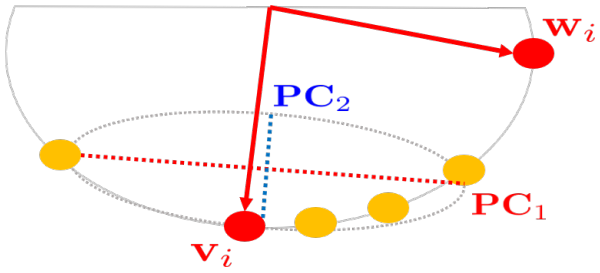


Figure 6.6: The yellow discs represent the unit length headings of agent i and its neighbors. The unit length vectors $\mathbf{w}_i$ and $\mathbf{v}_i$ are the principal vectors corresponding to the largest and smallest principal components PC1 and PC2 as seen by agent i.

6.3.2 The Swarming Algorithms

The PCA flow (6.3) as illustrated in Fig. 6.5 captures any change in the spatial geometry of the agents. A variation on the spatial geometry could be due to agents changing their relative speeds or changing their set of neighbors. On the other hand, the PCA flow (6.5) as illustrated in Fig. 6.6 captures any change in the headings of the neighboring agents. This motivates us to the following control law

$$\dot{\mathbf{r}}_i = (\frac{\lambda_i^q}{\lambda_i^q + \lambda_i^n})\alpha_i(\mathbf{s}_i, \mathbf{n}_i)\mathbf{n}_i + (\frac{\lambda_i^q}{\lambda_i^q + \lambda_i^n})\beta_i(\mathbf{s}_i, \mathbf{n}_i)\mathbf{v}_i \tag{6.6}$$

$$\dot{\mathbf{s}}_i = k_s(\mathbf{n}_i - \langle \mathbf{n}_i, \mathbf{s}_i \rangle \mathbf{s}_i), \tag{6.7}$$

where (6.7) is a consensus-on-a sphere control law that tracks the change of the PCA direction $\mathbf{n}_i$ from one-time step to another. The term $\frac{\lambda_i^q}{\lambda_i^q+\lambda_i^n}$ in (6.6) represents the shape of the swarm as seen by the i^{th} agent, which acts as indirect feedback signal to detect any disruption in the swarm shape. Note that $\frac{\lambda_i^q}{\lambda_i^q+\lambda_i^n} \in [0.5, 1]$ where it approaches 0.5 when the shape is circular, and it is grater than 0.5 when the shape is elongated. Let

$$\alpha_i = 1 - \langle \mathbf{n}_i, \mathbf{s}_i \rangle + \zeta_i, \qquad \beta_i = 1 + \langle \mathbf{n}_i, \mathbf{s}_i \rangle, \tag{6.8}$$

where ζ_i is a positive value that represents the signal that needs to be propagated to the entire swarm to achieve one of the scenarios of Figs. 6.1, 6.2 and 6.3. Note that $\beta_i =$

$2 - \alpha_i + \zeta_i$. Therefore, if $\zeta_i = 0$ and $\mathbf{s}_i = \mathbf{n}_i$, i.e. no change on the shape, then $\alpha_i = 0$ while $\beta_i = 2$. This implies that agent i is just aligning with its neighbors. On the other hand, when $\zeta_i \neq 0$ for some i, then the shape will change and hence $\mathbf{s}_i \neq \mathbf{n}_i$. In this case, $\alpha_i \neq 0$ and $\beta_i < 2$ which means that the control law is paying attention to the change of the shape. Note that in (6.6) there is no formation control part, which leads to a swarm that dynamically varies in shape according to the agents' speed and environment. Additionally, this control law does not ensure collision avoidance which might be considered using lower level controllers such as barrier functions as in [70].

6.4 PCA on Headings v.s. Average Consensus

In this section we make comparison between the PCA on headings (6.5) and the following consensus-on-a sphere control law

$$\dot{\mathbf{v}}_i = \sum_{j \in \mathcal{N}_i} (\mathbf{v}_j - \langle \mathbf{v}_i, \mathbf{v}_j \rangle \mathbf{v}_i) = \sum_{j \in \mathcal{N}_i} (\mathbf{I} - \mathbf{v}_i \mathbf{v}_i^T) \mathbf{v}_j, \tag{6.9}$$

where $\mathbf{v}_i$ is a vector of unit length representing the heading of agent i, and $\mathbf{I}$ is a 2 identity matrix. It is known that the consensus law (6.9) converges to the average value of the headings [21, 51]. Remark that the main difference between the average consensus (6.9) and the PCA on headings (6.5) is that the latter is weighted by the covariance matrix $\mathbf{H}_i$. In Fig. 6.7, we plot the result of the average consensus (in red) and the PCA heading consensus (in cyan). The obvious difference is that the PCA heading consensus tends to shift the result from the exact average towards the outlier heading. If the outlier heading is a piece of important information such as the case in predator avoidance, then the PCA heading consensus propagates this information faster than the average consensus especially in large networks where any outlying information is easily suppressed by averaging.

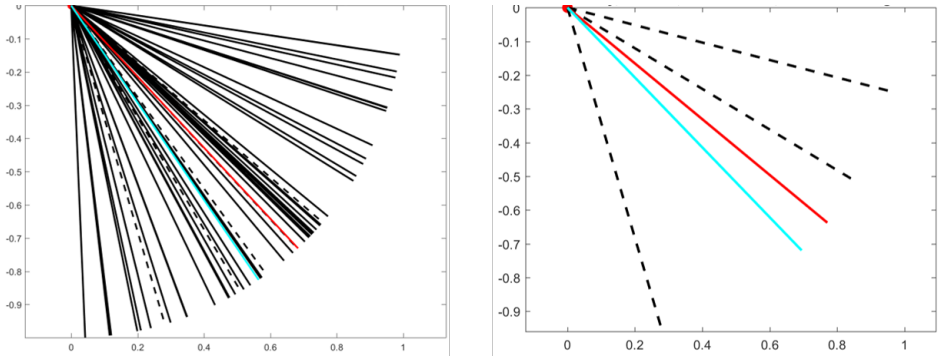


Figure 6.7: Heading vectors of a swarm of 50 agents connected by a Delaunay graph on the left, and a swarm of 4 agents connected by a complete graph on the right. The red is the average consensus while the cyan is the PCA heading consensus. The black lines represent the initial headings of the agents where they are dashed if they are neighbors to agent i and solid otherwise.

6.4.1 The Propagated Information

Now we show how to design the information ζ_i such that the swarm may exhibit the behaviors depicted in Figs. 6.1, 6.2 and 6.3.

For the predator avoidance, we design

$$\zeta_i = \begin{cases} \mu, & \text{if } ||\mathbf{r}_i - \mathbf{r}_{\text{predator}}|| < d^* \\ 0, & \text{otherwise} \end{cases} \tag{6.10}$$

where $\mu > 0$ is a speed constant, and d^* is the range of distance an agent can detect a predator. In view of the control law (6.6), when $\zeta_i = \mu$, agent i has a velocity component in the $\mathbf{n}_i$ direction. This leads to change on the local shape composed by the spatial distribution of the surrounding agents. Although the neighboring agents may have $\zeta_j = 0$, they still detect the change since their α_j in (6.8) is activated by the term $\langle \mathbf{n}_j, \mathbf{s}_j \rangle \neq 1$. Eventually, when all agents have $\zeta_i = 0$, then $\alpha_i = 0$ for all agents, and hence they keep moving in a synchronized motion.

On the other hand, if agent j wants to initiate a turning wave, then we design

$$\zeta_i = \begin{cases} \mu, & \text{if } i = j \\ 0, & \text{otherwise} \end{cases} \tag{6.11}$$

where $\zeta_i = \mu$ for an adequate time that changes deform the shape of the swarm and hence initiate a change on the local PCA on positions.

Finally, for the predator avoidance while performing source seeking, we design

$$\zeta_i = \begin{cases} k * z_i(\mathbf{r}_i), & \text{if } ||\mathbf{r}_i - \mathbf{r}_{\text{predator}}|| < d^* \\ z_i(\mathbf{r}_i), & \text{otherwise} \end{cases} \tag{6.12}$$

where $z_i(\mathbf{r}_i)$ is the field value as measured by agent i at its current position, and $k > 1$ is a constant. As the predator approaches the swarm, the agents who detect it increase their speed by a factor of k. This causes the shape to deform which in turn leads to a change of direction that propagates to the entire swarm.

Remark 6. *When for all agents $\zeta_i = z_i(\mathbf{r}_i)$, then the swarm is performing the source seeking behavior described in Chapter 4. However, the strategy now includes a PCA on headings term which synchronizes the motion direction even when the local PCA on positions produces significantly different $\mathbf{n}_i$ for each agent. It worth to mention that the cost for doing this is **Assumption** 6.2.3 which was not required before.*

6.5 Stability Analysis

In this section, we show that the individual signals ζ_i of (6.8) are implicitly propagated. We also show that the spatial variances of the shape of the swarm are bounded by their initial values, which ensures maintaining connectivity of the swarm. Although all the control algorithms are designed to work for any connected graph, in this section we only show the analysis of the complete graph. Analyzing the incomplete graphs, especially when they are

dynamic, is much harder and requires more treatment that will be considered in the future.

When the graph is complete, then all agents have the same covariance matrix

$$\mathbf{C}_i = \mathbf{C} = \sum_{i=1}^{M} (\mathbf{r}_i - \mathbf{r}_c)(\mathbf{r}_i - \mathbf{r}_c)^T. \tag{6.13}$$

This implies that all agents at a given time compute the same PCA components $\mathbf{n}_i = \mathbf{n}$, $\mathbf{q}_i = \mathbf{q}$, $\lambda_i^n = \lambda^n$ and $\lambda_i^q = \lambda^q$. Assume that each agent initiates its heading according to $\mathbf{h}_i(0) = \mathbf{n}_i(0)$. This implies that $\mathbf{v}_i = \mathbf{n}_i$, and hence we obtain from (6.6)

$$\dot{\mathbf{r}}_i = (2 + \zeta_i)(\frac{\lambda^q}{\lambda^q + \lambda^n})\mathbf{n}, \tag{6.14}$$

where we use $\alpha + \beta = 2 + \zeta_i$. Note that since (6.14) is independent of $\langle \mathbf{n}_i, \mathbf{s}_i \rangle$, then for complete graphs we don't need the tracking dynamics (6.7).

Recall that the PCA flow (6.3) runs in the time scale τ, while the swarming control law (6.14) runs in the time scale t. That is, for each time instance t, each agent runs (6.3) for some time τ. To overcome this difficulty, we formulate a singular perturbation problem. Let $\frac{dt}{d\tau} = \epsilon$ where $\epsilon \in (0, 1)$. This implies that $\tau = \frac{t - t_0}{\epsilon}$ where $\tau_0 = 0$. Hence, the dynamics of the problem are

$$\dot{\mathbf{r}}_i = (2 + \zeta_i)(\frac{\lambda^q}{\lambda^q + \lambda^n})\mathbf{n}, \tag{6.15}$$

$$\epsilon \frac{d\mathbf{q}}{dt} = (\mathbf{I} - \mathbf{q}(\tau)\mathbf{q}^T(\tau))\mathbf{C}(t)\mathbf{q}(\tau). \tag{6.16}$$

In the singular perturbation framework, we view (6.15) as a slow system, and (6.16) as a fast system. In this framework, we first let $\epsilon = 0$ in (6.16) to obtain decoupled reduced and boundary systems. We then analyze the stability of the origins of these decoupled systems. Finally, we derive $\epsilon^* \in (0, 1)$ such that for all $\epsilon \leq \epsilon^*$, the stability results of the reduced and boundary systems hold for the original slow (6.15) and fast (6.16) systems.

6.5.1 Derivation and Analysis of the Reduced System

When $\epsilon = 0$, the roots of the fast system (6.16) are the eigenvectors $(\mathbf{q}(t), \mathbf{n}(t))$ of the covariance matrix (6.13). Taking the time derivative of (6.13)

$$\dot{\mathbf{C}} = \sum_i (\dot{\mathbf{r}}_i - \dot{\mathbf{r}}_c)(\mathbf{r}_i - \mathbf{r}_c)^T + (\mathbf{r}_i - \mathbf{r}_c)(\dot{\mathbf{r}}_i - \dot{\mathbf{r}}_c)^T \tag{6.17}$$

Recall that $\mathbf{r}_c = \frac{1}{M}\sum_i \mathbf{r}_i$. Using (6.14), we obtain

$$\dot{\mathbf{r}}_c = \frac{1}{M}\sum_i \dot{\mathbf{r}}_i = (2 + \zeta_a)(\frac{\lambda^q}{\lambda^q + \lambda^n})\mathbf{n}, \tag{6.18}$$

where $\zeta_a = \frac{1}{M}\sum_{i=1}^{M} \zeta_i$. Then, using (6.14) and (6.18)

$$\dot{\mathbf{r}}_i - \dot{\mathbf{r}}_c = (\frac{\lambda^q}{\lambda^q + \lambda^n})(\zeta_i - \zeta_a)\mathbf{n}. \tag{6.19}$$

Plugging (6.19) in (6.17) to obtain

$$\dot{\mathbf{C}} = (\frac{\lambda^q}{\lambda^q + \lambda^n})\sum_i (\zeta_i - \zeta_a)[\mathbf{n}(\mathbf{r}_i - \mathbf{r}_c)^T + (\mathbf{r}_i - \mathbf{r}_c)\mathbf{n}^T] \tag{6.20}$$

Since the covariance matrix satisfies $\mathbf{C}\mathbf{n} = \lambda^n \mathbf{n}$, then

$$\dot{\mathbf{C}}\mathbf{n} + \mathbf{C}\dot{\mathbf{n}} = \dot{\lambda}^n \mathbf{n} + \lambda^n \dot{\mathbf{n}}, \tag{6.21}$$

Inner product both sides of (6.21) with the eigenvector $\mathbf{q}$

$$\langle \mathbf{q}, \dot{\mathbf{C}}\mathbf{n}\rangle + \langle \mathbf{q}, \mathbf{C}\dot{\mathbf{n}}\rangle = \dot{\lambda}^n \langle \mathbf{q}, \mathbf{n}\rangle + \lambda^n \langle \mathbf{q}, \dot{\mathbf{n}}\rangle. \tag{6.22}$$

Using the fact that $\langle \mathbf{q}, \mathbf{n} \rangle = 0$, and $\mathbf{C}\mathbf{q} = \lambda^q \mathbf{q}$, (6.22) simplifies to

$$\langle \mathbf{q}, \dot{\mathbf{C}}\mathbf{n} \rangle = (\lambda^n - \lambda^q) \langle \mathbf{q}, \dot{\mathbf{n}} \rangle. \tag{6.23}$$

But, using (6.20) and since $\langle \mathbf{q}, \mathbf{n} \rangle = 0$, we also obtain

$$\langle \mathbf{q}, \dot{\mathbf{C}}\mathbf{n} \rangle = (\frac{\lambda^q}{\lambda^q + \lambda^n}) \sum_i (\zeta_i - \zeta_a) \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{q} \rangle, \tag{6.24}$$

Equating (6.23) and (6.24), and for $\lambda^n - \lambda^q \neq 0$, we obtain

$$\langle \mathbf{q}, \dot{\mathbf{n}} \rangle = -\frac{\lambda^q}{(\lambda^q - \lambda^n)(\lambda^q + \lambda^n)} \sum_i (\zeta_i - \zeta_a) \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{q} \rangle \tag{6.25}$$

Similarly, starting from $\mathbf{C}\mathbf{q} = \lambda^q \mathbf{q}$ and following the same procedure of (6.21) to (6.24), we obtain

$$\langle \mathbf{n}, \dot{\mathbf{q}} \rangle = \frac{\lambda^q}{(\lambda^q - \lambda^n)(\lambda^q + \lambda^n)} \sum_i (\zeta_i - \zeta_a) \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{q} \rangle \tag{6.26}$$

Since $\mathbf{n}$ and $\mathbf{q}$ are orthonormal, then we can write

$$\dot{\mathbf{n}} = \langle \mathbf{q}, \dot{\mathbf{n}} \rangle \mathbf{q}, \quad \dot{\mathbf{q}} = \langle \mathbf{n}, \dot{\mathbf{q}} \rangle \mathbf{n}. \tag{6.27}$$

Substituting (6.25) and (6.26) into (6.27), we obtain the dynamics

$$\dot{\mathbf{n}} = \kappa \mathbf{q}, \qquad \dot{\mathbf{q}} = -\kappa \mathbf{n}, \tag{6.28}$$

where the curvature of the trajectory κ, which also can be viewed as the angular velocity of the swarm, is given by

$$\kappa = \frac{\lambda^q}{(\lambda^q - \lambda^n)(\lambda^q + \lambda^n)} \sum_i (\zeta_a - \zeta_i) \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{q} \rangle. \tag{6.29}$$

Before proceeding, we make the following observations:

1. Since κ is a function of all ζ_i, it implies that the individual speeds are implicitly propagated.

2. The curvature is zero when all agents have the same ζ_i. This implies that the swarm is moving in a straight line.

3. The curvature is non-zero when one or more agents have different $\zeta_i \ = 0$. This implies that the swarm will turn clockwise or counterclockwise depending on ζ_i as well as on the location of the i^{th} agent with respect to the center of the swarm. Hence, any agent can change the swarm direction by just varying its speed.

The spatial variances of the swarm are represented by $\lambda^n = \sum_i \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n} \rangle^2$ and $\lambda^q = \sum_i \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{q} \rangle^2$. Taking the time derivative of λ^n

$$\dot{\lambda}^n = 2 \sum_i \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n} \rangle [\langle \dot{\mathbf{r}}_i - \dot{\mathbf{r}}_c, \mathbf{n} \rangle + \langle \mathbf{r}_i - \mathbf{r}_c, \dot{\mathbf{n}} \rangle] \tag{6.30}$$

Using (6.19) and (6.28)

$$\begin{aligned} \dot{\lambda}^n &= 2 \sum_i \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n} \rangle [\frac{\lambda^q}{\lambda^q + \lambda^n} (\zeta_i - \zeta_a) + \langle \mathbf{r}_i - \mathbf{r}_c, \kappa \mathbf{q} \rangle] \\ &= 2 \frac{\lambda^q}{\lambda^q + \lambda^n} \sum_i (\zeta_i - \zeta_a) \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n} \rangle \\ &+ 2 \kappa \mathbf{n}^T \sum_i (\mathbf{r}_i - \mathbf{r}_c)(\mathbf{r}_i - \mathbf{r}_c)^T \mathbf{q} \end{aligned} \tag{6.31}$$

But $\mathbf{n}^T \sum_i (\mathbf{r}_i - \mathbf{r}_c)(\mathbf{r}_i - \mathbf{r}_c)^T \mathbf{q} = \mathbf{n}^T \mathbf{C} \mathbf{q} = \lambda^q \mathbf{n}^T \mathbf{q} = 0$. Hence

$$\dot{\lambda}^n = 2 \frac{\lambda^q}{\lambda^q + \lambda^n} \sum_i (\zeta_i - \zeta_a) \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n} \rangle \tag{6.32}$$

Similarly, taking the time derivative of λ^n

$$\dot{\lambda}^q = 2\sum_i \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{q}\rangle[\langle \dot{\mathbf{r}}_i - \dot{\mathbf{r}}_c, \mathbf{q}\rangle + \langle \mathbf{r}_i - \mathbf{r}_c, \dot{\mathbf{q}}\rangle] \tag{6.33}$$

Using (6.19) and (6.28)

$$\begin{aligned}\dot{\lambda}^q = 2\sum_i \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{q}\rangle[\frac{\lambda^q}{\lambda^q + \lambda^n}(\zeta_i - \zeta_a)\langle \mathbf{n}, \mathbf{q}\rangle \\ - \langle \mathbf{r}_i - \mathbf{r}_c, \kappa\mathbf{n}\rangle] = -2\kappa\mathbf{q}^T\mathbf{C}\mathbf{n} = -2\kappa\lambda^n\mathbf{q}^T\mathbf{n} = 0.\end{aligned} \tag{6.34}$$

Equations (6.32) and (6.34) represent the reduced system representing the dynamics of the spatial variances of the shape of the swarm when the PCA flow (6.16) converges instantly. We then prove the following result.

Lemma 6.5.1. *Consider the system given by (6.15) and (6.16). When* $\epsilon = 0$, $\lambda^n(t) \leq \lambda^q(t) = \lambda^q(t_0)$, *where* $\lambda^q(t_0)$ *is the initial maximum variance of the spatial distribution of the agents.*

Proof. From (6.34), we conclude that λ^q is constant, i.e. equals to the initial value. On the other hand, from (6.32) we see that λ^n may increase or decrease depending on the sign of $(\zeta_i - \zeta_a)\langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n}\rangle$. However, from the property of PCA, we know that it may increase only up to $\lambda^n = \lambda^q$ where at that time, they interchange. Hence λ^n is always bounded by λ^q. □

6.5.2 Derivation and Analysis of the Boundary System

Let $\psi = 1 - \langle \mathbf{q}(t), \mathbf{q}(\tau)\rangle$ where $\psi = 0$ if and only if $\mathbf{q}(t) = \mathbf{q}(\tau)$. Then the dynamics of the fast system are:

$$\frac{d\psi}{d\tau} = -\langle \frac{d\mathbf{q}(t)}{d\tau}, \mathbf{q}(\tau)\rangle - \langle \mathbf{q}(t), \frac{d\mathbf{q}(\tau)}{d\tau}\rangle \tag{6.35}$$

By the Chain rule, $\frac{d\mathbf{q}(t)}{d\tau} = \epsilon \frac{d\mathbf{q}(t)}{dt}$ and $\frac{d\mathbf{q}(\tau)}{dt} = \frac{1}{\epsilon}\frac{d\mathbf{q}(\tau)}{d\tau}$, and hence:

$$\frac{d\psi}{d\tau} = -\epsilon\langle\frac{d\mathbf{q}(t)}{dt}, \mathbf{q}(\tau)\rangle - \langle\mathbf{q}(t), \frac{d\mathbf{q}(\tau)}{d\tau}\rangle. \tag{6.36}$$

$$\frac{d\psi}{dt} = -\langle\frac{d\mathbf{q}(t)}{dt}, \mathbf{q}(\tau)\rangle - \frac{1}{\epsilon}\langle\mathbf{q}(t), \frac{d\mathbf{q}(\tau)}{d\tau}\rangle. \tag{6.37}$$

Setting $\epsilon = 0$ in (6.36) and using (6.3), we obtain the boundary system dynamics:

$$\frac{d\psi}{d\tau} = (1-\psi)(\langle\mathbf{q}(\tau), \mathbf{C}(t)\mathbf{q}(\tau)\rangle - \lambda^q). \tag{6.38}$$

Observe that in (6.38), since $\epsilon = 0$ then $\mathbf{q}(t)$ and $\mathbf{C}(t)$ are constants with respect to the time scale τ.

Lemma 6.5.2. *Consider (6.38). Suppose that at time $\tau = 0$, $\langle\mathbf{q}(t), \mathbf{q}(\tau)\rangle \in (0, 1]$. Then the origin of the boundary system is exponentially stable uniformly in $\mathbf{C}(t)$ and $\mathbf{q}(t)$.*

This lemma is similar to **Lemma** 4.5.2 of Chapter 4 with the proof therein.

6.5.3 When $\epsilon \neq 0$

We are going to follow the same procedure of deriving (6.32) and (6.34). However, here we use (6.16) instead of (6.28). After several manipulations, we can derive

$$\begin{aligned}\dot{\lambda}^n = &2\frac{\lambda^q}{\lambda^q + \lambda^n}\sum_i(\zeta_i - \zeta_a)\langle\mathbf{r}_i - \mathbf{r}_c, \mathbf{n}(t)\rangle \\ &+ \frac{2\lambda^n}{\epsilon}[\lambda^q - \mathbf{q}^T(\tau)\mathbf{C}(t)\mathbf{q}(\tau)]\langle\mathbf{q}(t), \mathbf{q}(\tau)\rangle\end{aligned} \tag{6.39}$$

and

$$\dot{\lambda}^q = \frac{2}{\epsilon}\lambda^q[\lambda^q - \mathbf{q}^T(\tau)\mathbf{C}(t)\mathbf{q}(\tau)]\langle\mathbf{q}(t), \mathbf{q}(\tau)\rangle \tag{6.40}$$

Theorem 6.5.1. *Consider (6.15) and (6.16). Suppose that at time* $\tau = 0$, $\langle \mathbf{q}(t), \mathbf{q}(\tau) \rangle \in (0, 1]$. *Then the trajectories of* λ^q *and* λ^n *are ultimately bounded* $\forall\ \epsilon < \frac{2\gamma(1+4\lambda^q)}{\sum_i |(\zeta_i - \zeta_a)|}$ *where* $\gamma \in (0, 1)$.

Proof. Let $V = \lambda^q + \lambda^n + \psi$ be a Lyapunov candidate function. Using (6.39), (6.40), and (6.37), we obtain

$$\begin{aligned}
\dot{V} &= 2\frac{\lambda^q}{\lambda^q + \lambda^n} \sum_i (\zeta_i - \zeta_a) \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n}(t) \rangle \\
&+ \frac{2}{\epsilon}[\lambda^q - \mathbf{q}^T(\tau)\mathbf{C}(t)\mathbf{q}(\tau)](1 - \psi)(\lambda^q + \lambda^n) - \frac{1}{\epsilon}[\lambda^q - \mathbf{q}^T(\tau)\mathbf{C}(t)\mathbf{q}(\tau)] \\
&= 2\frac{\lambda^q}{\lambda^q + \lambda^n} \sum_i (\zeta_i - \zeta_a) \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n}(t) \rangle \\
&+ \frac{1}{\epsilon}[\lambda^q - \mathbf{q}^T(\tau)\mathbf{C}(t)\mathbf{q}(\tau)][2(\lambda^q + \lambda^n)(1 - \psi) - 1] \qquad (6.41)
\end{aligned}$$

Since $\mathbf{q}^T(\tau)\mathbf{C}(t)\mathbf{q}(\tau) \leq \lambda_{max}(\mathbf{C}(t)) = \lambda^q$, then whenever $\psi > 1$, we have

$$W = [\lambda^q - \mathbf{q}^T(\tau)\mathbf{C}(t)\mathbf{q}(\tau)][1 - 2(\lambda^q + \lambda^n)(1 - \psi)] \qquad (6.42)$$

is a continuous positive function. Additionally

$$2\frac{\lambda^q}{\lambda^q + \lambda^n} |\sum_i (\zeta_i - \zeta_a) \langle \mathbf{r}_i - \mathbf{r}_c, \mathbf{n}(t) \rangle| \leq \frac{(\lambda^q)^2}{\lambda^q + \lambda^n} \sum_i |(\zeta_i - \zeta_a)|. \qquad (6.43)$$

We can then show that

$$\dot{V} \leq -(1 - \gamma)W, \quad \forall \sqrt{(\lambda^q)^2 + (\lambda^n)^2 + \psi^2} > 1, \qquad (6.44)$$

and $\forall\ \epsilon < \frac{2\gamma(1+4\lambda^q)}{\sum_i |(\zeta_i - \zeta_a)|}$ where $\gamma \in (0, 1)$. This along with Lemma 6.5.1 and Lemma 6.5.2 imply that the trajectories of λ^q and λ^n using the singularly perturbed systems (6.15) and (6.16) are ultimately bounded $\forall\ \epsilon < \frac{2\gamma(1+4\lambda^q)}{\sum_i |(\zeta_i - \zeta_a)|}$. □

6.6 Simulation Results

In this section, we simulate the proposed control algorithms for various scenarios. In all the simulations, we set $\epsilon = 0.01$. The red discs denote the agents while the blue lines denote the edges of the graph. The red arrows represent the headings of each agent. The black diamond is the predator, which moves in a circular motion.

In Fig. 6.8, we demonstrate a predator avoidance behavior for a swarm of 4 agents using a complete graph. In this case, we set $\zeta_i = 5$ if $||\mathbf{r}_i - \mathbf{r}_{\text{predator}}|| < 0.5$ and $\zeta_i = 0$ otherwise. The swarm is moving in a straight line until, at time $t = t_1$, one agent detects the predator. This agent increases its speed which causes a change in the heading of the swarm. Note that only one agent detects the predator, but all agents respond instantly as all of them immediately detect the change of the shape of the swarm. Fig. 6.9, we repeat the same simulation but for 6 agents.

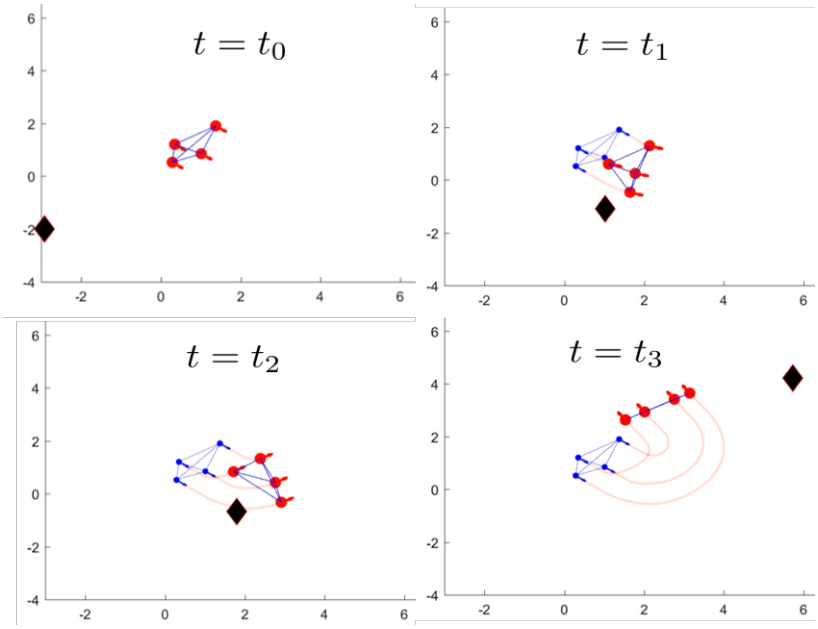


Figure 6.8: TB

Alternatively, in Fig. 6.10 and Fig. 6.11, two swarms start at the same initial positions and their agents interact with each other using a Delaunay graph. The difference is the agent who initiates the turn which is emphasized by a green circle. This agent increases its

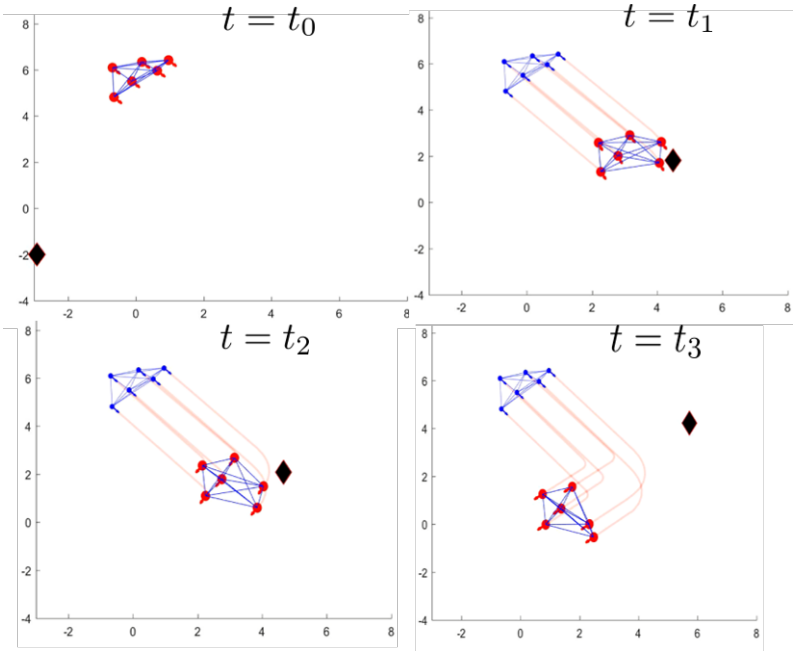


Figure 6.9: Predator avoidance of a swarm of six agents using a complete graph. The agent who detects the predator increases its speed causing the whole swarm to immediately turn.

speed for a short period of time that disrupts the shape of the swarm, creating an implicit turning signal that propagates to the entire swarm. Depending on the location of the agent who initiates the signal, each swarm ends up turning to a different direction.

All these simulation results suggest the effectiveness of the proposed model for various collective behaviors that require information propagation for successful multi-tasking. The agents using simple interaction rules with only local information are able to detect and respond to various signals initiated as a reaction to the environment or external threats.

To demonstrate the sequence of the propagating wave between the agents, in Fig. 6.13, we plot the angle of the heading of the six agents of the swarm shown in Fig. 6.12 with respect to the positive x-axis. Between times $t = 1$ and $t = 1.3$, agent three in Fig. 6.12 increases its speed, leading to a turning wave. In this time period, we see in Fig. 6.13 that the propagation sequence follows the graph structure. Note that agent six starts turning before agent one, which is closer to agent three than agent six. This is related to the graph structure as well as to the location with respect to the swarm.

To validate the scalability of the proposed model to swarms of a large number of agents,

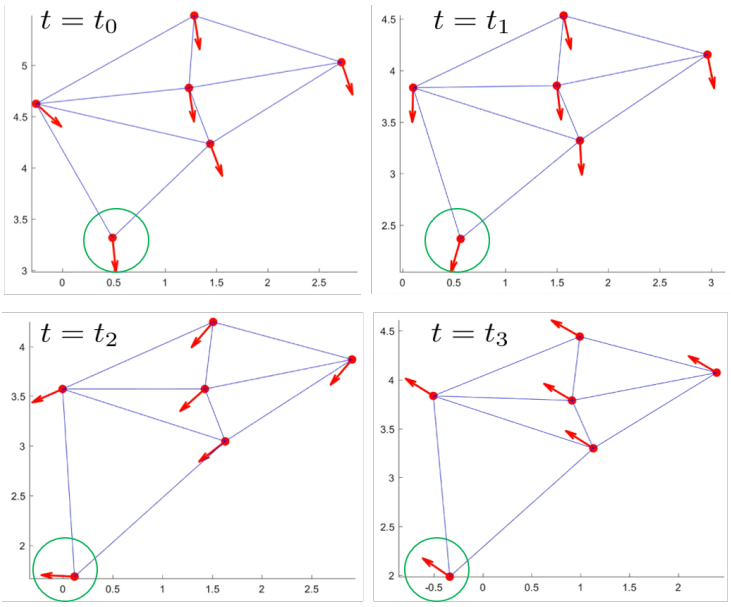


Figure 6.10: A swarm of six agents that interact with each other through a Delaunay graph. The green circle emphasizes the agent who initiated the turning wave. Note that the scale of the figure changes from one time to another as the swarm navigates.

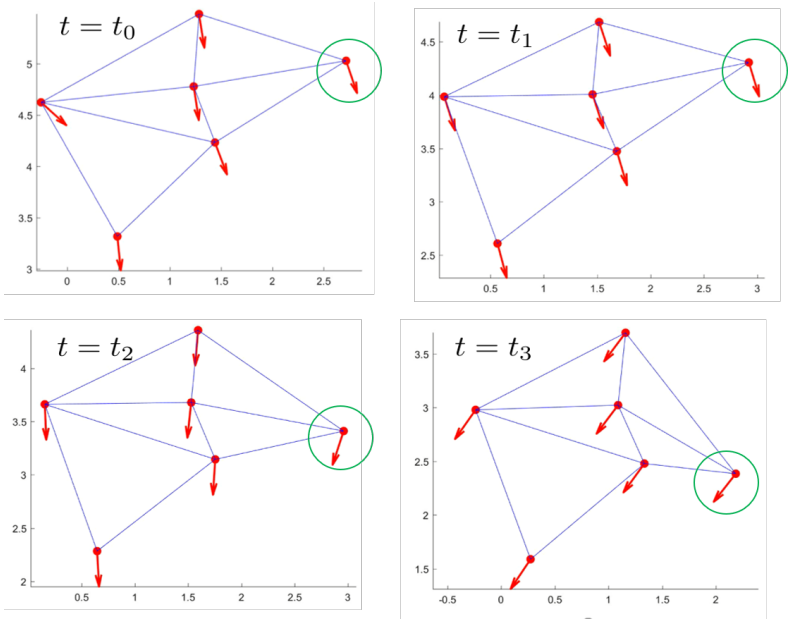


Figure 6.11: A swarm of six agents that interact with each other through a Delaunay graph. The green circle emphasizes the agent who initiated the turning wave. Note that the scale of the figure changes from one time to another as the swarm navigates.

we simulate it for a swarm of 20 agents in Fig. 6.14. Although the swarm is large, one agent is able to initiate a turning wave that propagates quickly to the entire swarm. Remark

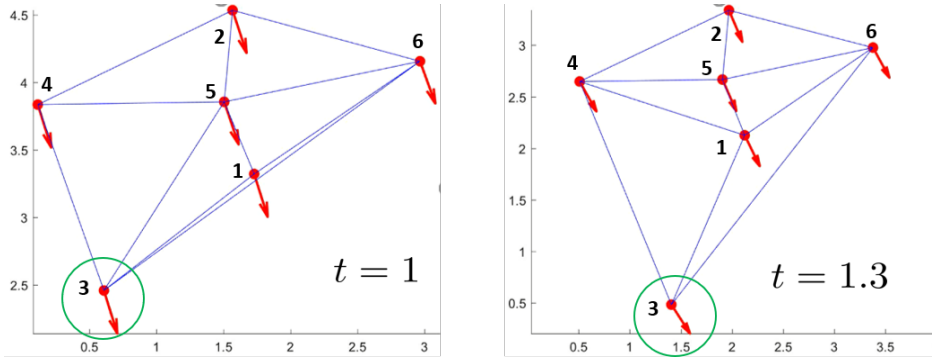


Figure 6.12: At time $t = 1$, agent three increase its speed until time $t = 1.3$. In Fig. 6.13 we show the sequence of the propagation of the turning wave.

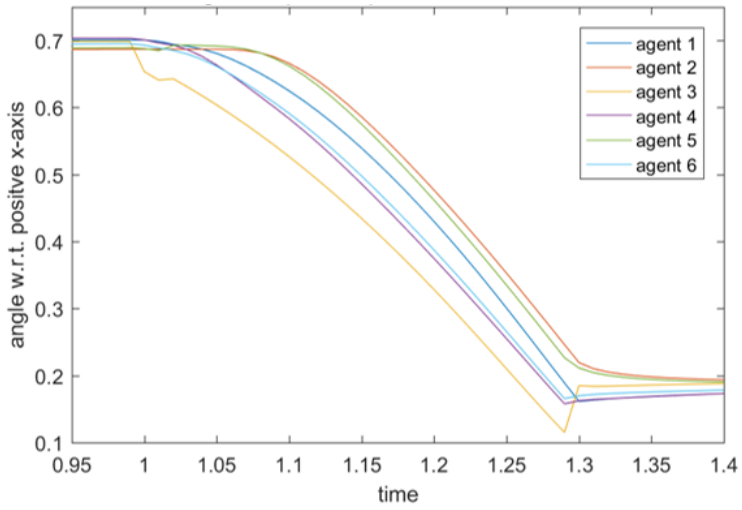


Figure 6.13: These curves represent the heading angle of the six agents from Fig. 6.12 with respect to the positive x-axis. Clearly, the turning sequence follows the graph structure.

that the agent who initiated the turn was initially in front of the swarm. After the turn, it becomes at the back of the swarm.

To further justify the capability of the proposed model in achieving information propagation and multi-tasking, we simulate it in Fig. 6.15 for a swarm of 8 agents collectively performing distributed Speeding Up and Slowing Down source seeking strategy [8]. At the same time, there is a predator that the agents need to avoid while seeking the source. In this figure, the contour lines represent the level curves of the field which is set to be a quadratic function. The source is located at the origin where the field has a minimum value. In this

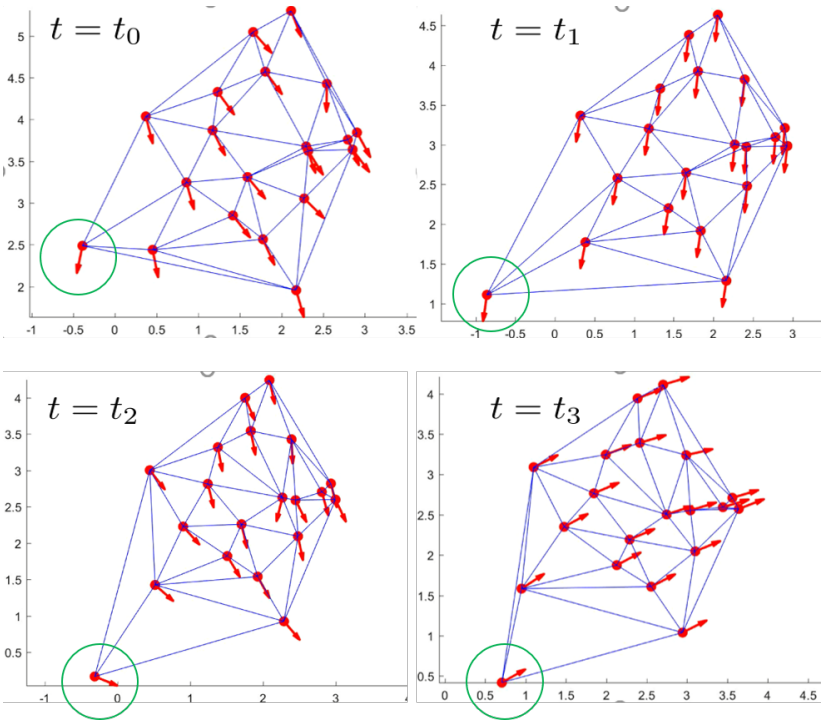


Figure 6.14: The circled agent successfully enforces the other 19 agents of the swarm to reverse their headings to the opposite direction by simply modulating its speed.

scenario, we set the speed to be

$$\zeta_i = \begin{cases} 5z_i(\mathbf{r}_i) & : ||\mathbf{r}_i - \mathbf{r}_{\text{predator}}|| < 2 \\ z_i(\mathbf{r}_i) & : \text{otherwise} \end{cases} \tag{6.45}$$

where $z_i(\mathbf{r}_i)$ is the field value as measured by agent i at its current position. As the predator approaches the swarm, the agents who detect it increase their speed leading to a change of direction that propagates to the entire swarm. The swarm successfully escapes from the predator and autonomously returns to its source seeking task.

6.7 CONCLUSIONS

In this chapter, we innovated a Multi-Layer control model composed of an interplay of decentralized algorithms for perception and swarming. Through this novel model, we demonstrate implicit information propagation and multi-tasking in swarms using only local

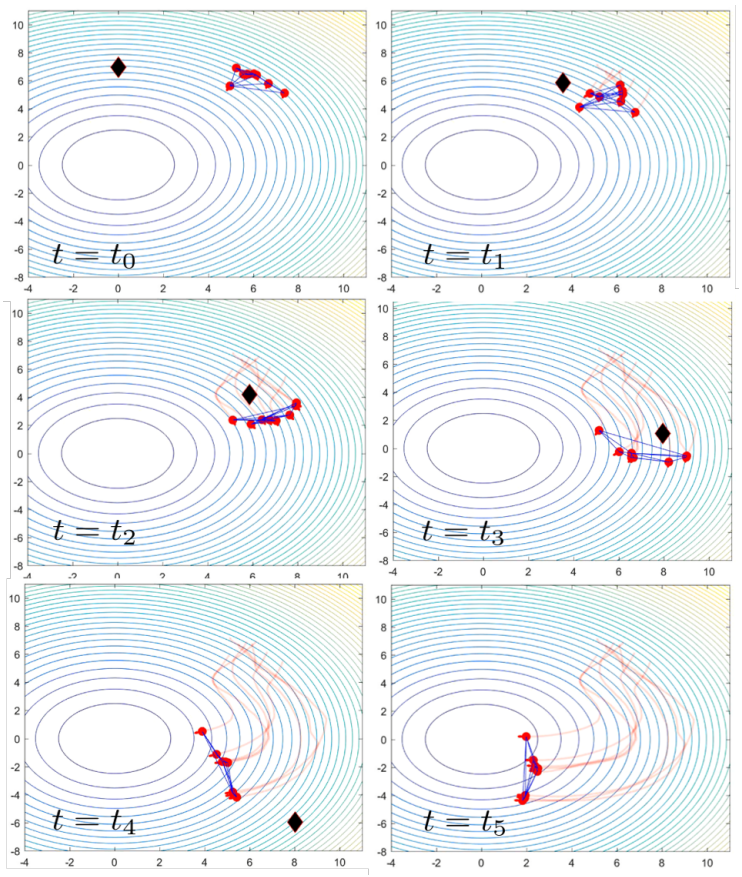


Figure 6.15: A swarm of eight agents successfully avoid a predator while collectively performing source seeking. The contours represent the level curves of the field which is minimum at the origin.

interactions and without explicit communication or prescribed formations. We prove that variations on individual speed are implicitly propagated across the swarm, and we prove that the spatial variances of the shape of the swarm are bounded, which implicitly ensures the connectivity of the graph. Finally, we provide various simulation results demonstrating the effectiveness of the model for swarms with complete and incomplete graphs performing collective synchronization and source seeking while avoiding a predator.

CHAPTER 7
CONCLUDING REMARKS AND FUTURE RESEARCH

7.1 Conclusion

The main contribution of this Dissertation is a Multi-Layer control model composed of an interplay of decentralized algorithms for perception and swarming. In the perception layer, each agent applies a Principal Component Analysis (PCA) on the relative positions and headings of its neighbors to learn principal properties about the motion and the geometry of the spatial distribution of the surrounding agents. These principal components are then used in the swarming layer where various distributed control laws are designed to balance between achieving a collective task and at the same time allowing critical emerging signals to propagate to the entire swarm.

The detailed contribution of this Dissertation is summarized as follows.

- Incorporating a consensus-on-a sphere to extend an existing 3-dimensional gradient-free source seeking strategy from a 3-agent system into M-agent system. For this, we developed a two-layer system composed of social and environmental layers. In the social layer, the agents interact with each other by means of implicit and explicit consensus and formation control laws. In the environmental layer, the agents interact with the environment by modulating their speed as a function of the field value.

- Solving the distributed source seeking problem for swarms of various sizes and graph structures without explicitly estimating the field gradient and without explicitly communicating field measurements or coordination components. For this, we developed a Multi-Layer model wherein the perception layer, each agent learns from the spatial distribution of its neighbors principal directions for motion. These directions are then used in the swarming layer to modulate velocities based on the environmental field

value.

- Solving the distributed level curve tracking problem for swarms of various sizes and graph structures without explicitly estimating the field gradient and without explicitly communicating field measurements or coordination components. For this, we modify the distributed control law in the swarming layer of the Multi-Layer model such that the swarm converges first to the desired level curve and then keeps tracking it.

- Modeling information propagation in swarms without explicit communication or prescribed formations. For this, we added an extra layer to the Multi-Layer model where in this each agent applies PCA on relative headings of its neighbors to extract geometric components about the motion of its neighbors. These components are then used in the swarming layer where a distributed control law is designed to balance between achieving a collective task and at the same time allowing critical emerging signals to propagate to the entire swarm.

- Obtaining various stability results that reflect the convergence and robustness of the proposed algorithms. Dealing with the multi-scale time complexity of the proposed algorithms, we utilized a singular perturbation framework to obtain the theoretical results. The main difficulty in obtaining these results lays in the fact that we are proving emergent behavior that we don't explicitly control. This requires complicated derivations and manipulations in order to obtain implicit dynamics that describe the desired emergent behaviors. Additionally, most of the existing source seeking and level curve tracking works ignore the higher order terms of the field. In contrast, in this Dissertation, when the graph is complete, we do consider the entire field components which allow us to realize their effect on the convergence and robustness of the algorithms.

- Validating the proposed model for numerous source seeking, level curve tracking and predator avoidance behaviors through various simulation and experimental re-

sults. The experiments are conducted using the Georgia Teach Robotarium and the Georgia Teach Miniature Blimp. These results suggest the efficiency, scalability, and robustness of the proposed model under different environments and robotic platforms.

7.2 Future Research

The proposed control model offers a new method that enables robots with limited resources to perform various swarming activities with only local information. The PCA algorithm used in the perception layer may be viewed as an engineering solution to facilitate information propagation. Designing analytical models to understand information propagation will not only reveal natural mysteries but additionally will help to propose multi-tasking control algorithms for robotic swarms that require only very limited or no explicit communication. In particular, this is highly related to the problem of designing tactics for a swarm of drones to avoid or chase a malicious agent [24, 25].

In biological research, evidence has been established to support the existence and necessity of individual differences in natural swarms, see for example [74, 75, 76, 77, 78]. These individual differences often cause the swarm to exhibit various collective behaviors when responding to internal or external stimuli. The differences might arise due to individuals with varying sensing, computing and motion capabilities [75]. Based on these variations, each individual may interact differently with the environment and their peers and may prefer certain locations in the swarm [76]. For example, it has been shown that individuals on the front or boundary of a swarm are more agile than those inside the swarm which supports certain swarm behaviors when attacked by a predator [79]. It has also been shown that critical swarm decisions in various behaviors are initiated by individuals located at the boundary of the swarm [3]. In some cases, individuals with higher capabilities tend to interact less with other individuals and hence often play a leader role [79].

Therefore, it might be more practical and rewarding to incorporate individual differ-

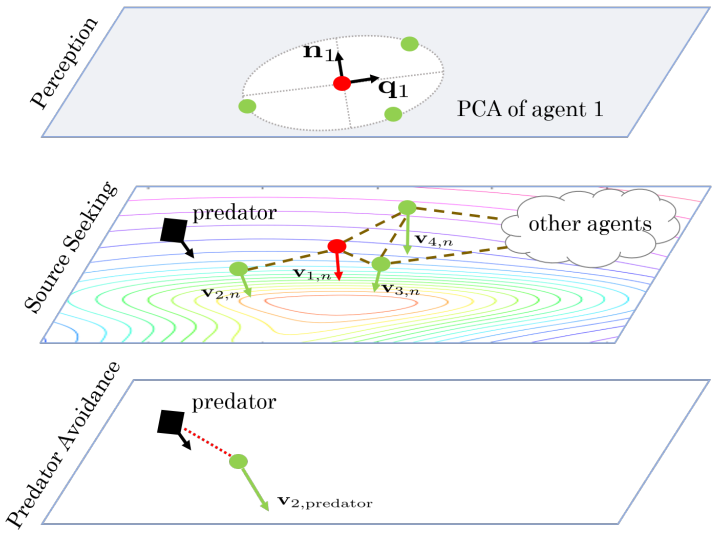


Figure 7.1: In the Perception and Source Seeking layers the agent interact with each other and the environment homogeneously. However, Only agents who recognize the predator react to it in the bottom layer.

ences in understanding information propagation across the swarm. In view of the proposed Multi-Layer model, the same individuals may be heterogeneous at one layer and homogeneous at another layer. For example, in a scenario of a swarm performing source seeking while at the same time avoiding predators as shown in Fig. 7.1, all the individuals respond to the environmental value in an identical mechanism. However, not all individuals are able to recognize a predator due to lack of capability or experience or due to their locations in the swarm. Therefore, few agents might be able to recognize a predator and exhibit the appropriate response that propagates to the entire swarm. Hence, we propose to add another layer that describes how individual differences characterize the swarm response to external predators or perturbations.

Since the PCA learning algorithm in the Perception layer depends on the capability of each agent to sense the relative positions of its neighbors, individual differences of sensing capabilities make each agent choose its set of neighbors differently. In [47], authors show that the efficiency of a swarm response to an external perturbation or predator increases

with the number of social interactions only up to a certain number, and then deteriorates. Various neighboring models describing how agents to access information from their neighbors are analyzed in [46, 47]. In particular, the neighbors are selected based on distance in the metric model, based on a number of agents in the topological model, and based on bearing angles in the visual model. Visual models are suitable for searching, while topological models are suitable for avoiding predators. Using the PCA Flow, we can alternatively specify the set of neighbors based on the spatial variances. Additionally, the location of each agent in the swarm (boundary or inside), may force each agent to select its neighbors differently to achieve a certain collective behavior. So it is worthwhile to study the effects of the individual differences on the PCA learning algorithm. For example, in a predator-avoidance behavior, when a predator removes an agent from the boundary of the swarm, it might cause a significant disruption in the principal directions more than removing an individual from the inside of the swarm. This can be viewed as an advantage in the sense that the swarm quickly detects the threat and makes the necessary decision.

REFERENCES

[1] D. T. Swain, I. D. Couzin, and N. E. Leonard, "Real-time feedback-controlled robotic fish for behavioral experiments with fish schools," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 150–163, 2012.

[2] A. Attanasi, A. Cavagna, L. D. Castello, I. Giardina, T. S. Grigera, A. Jeli, S. Melillo, L. Parisi, O. Pohl, E. Shen, and M. Viale, "Information transfer and behavioural inertia in starling flocks," *Nature Physics*, vol. 10, 691696, 2014.

[3] A. Attanasi, A. Cavagna, L. D. Castello, I. Giardina, A. Jelic, S. Melillo, L. Parisi, O. Pohl, E. Shen, and M. Viale, "Emergence of collective changes in travel direction of starling flocks from individual birds' fluctuations," *Journal of The Royal Society Interface*, vol. 12, no. 108, 2015.

[4] F. Rossi, S. Bandyopadhyay, M. Wolf, and M. Pavone, "Review of multi-agent algorithms for collective behavior: A structural taxonomy," *IFAC-PapersOnLine*, vol. 51, no. 12, pp. 112 –117, 2018, IFAC Workshop on Networked Autonomous Air Space Systems NAASS 2018.

[5] K. Oh, M. Park, and H. Ahn, "A survey of multi-agent formation control," *Automatica*, vol. 53, pp. 424 –440, 2015.

[6] Y. Mohan and S. G. Ponnambalam, "An extensive review of research in swarm robotics," in *World Congress on Nature and Biologically Inspired Computing (NaBIC)*, 2009, pp. 140 –145.

[7] M. Hutchinson, H. Oh, and W.-H. Chen, "A review of source term estimation methods for atmospheric dispersion events using static or mobile sensors," *Information Fusion*, vol. 36, pp. 130 –148, 2017.

[8] S. Al-Abri and F. Zhang, "Consensus on a sphere for a 3-dimensional speeding up and slowing down strategy," in *Proceedings of 56th IEEE Conference on Decision and Control*, 2017, pp. 1503 –1508.

[9] S. Al-Abri, W. Wu, and F. Zhang, "A gradient-free 3-dimensional source seeking strategy with robustness analysis," *IEEE Transactions on Automatic Control*, vol. 64, no. 8, 2019.

[10] S. Al-Abri, S. Maxon, and F. Zhang, "Integrating a pca learning algorithm with the susd strategy for a collective source seeking behavior," in *the Proceedings of the American Control Conference*, 2018, pp. 2479–2484.

[11] S. Al-Abri and F. Zhang, "A distributed level curve tracking control law for multi-agent systems," in *Proceedings of the 57th IEEE Conference on Decision and Control*, 2018, pp. 2575–2580.

[12] S. Al-Abri and F. Zhang, "A multi-layer model for a communication-free combined source seeking and level curve tracking," *IEEE Transactions on Automatic Control*, submitted.

[13] E. Oja, "Simplified neuron model as a principal component analyzer," *International Journal of Mathematical Biology*, vol. 15, no. 3, pp. 267–273, 1982.

[14] P. F. Baldi and K. Homik, "Learning in linear neural network: A survey," *IEEE Transactions on Neural Networks*, vol. 6, no. 4, pp. 837–858, 1995.

[15] U. H. S. Yoshizawa and K. Starkov, "Convergent analysis for principal component flows," *International Journal of Applied Mathematics and Computer Science*, vol. 11, no. 1, pp. 223–236, 2001.

[16] A. Berdahl, C. Torney, C. Ioannou, J. Faria, and I. Couzin, "Emergent sensing of complex environments by mobile animal groups," *Science*, vol. 339, no. 6119, pp. 574–576, 2013.

[17] S. Al-Abri, S. Maxon, and F. Zhang, "A multi-layer swarm control model for information propagation and multi-tasking," in *To appear in the Proceedings of the American Control Conference*, 2019.

[18] T. Vicsek and A. Zafeiris, "Collective motion," *Physics Reports*, vol. 517, no. 3, pp. 71 –140, 2012.

[19] A. Cavagna, L. D. Castello, I. Giardina, T. S. Grigera, A. Jelic, S. Melillo, T. Mora, L. Parisi, E. Silvestri, M. Viale, and A. M. Walczak, "Flocking and turning: A new model for self-organized collective motion," *CoRR*, vol. abs/1403.1202, 2014.

[20] A. Cavagna, I. Giardina, and T. S. Grigera, "The physics of flocking: Correlation as a compass from experiments to theory," *Physics Reports*, vol. 728, pp. 1 –62, 2018.

[21] J. Markdahl, J. Thunberg, and J. Gonalves, "Almost global consensus on the n-sphere," *IEEE Transactions on Automatic Control*, vol. 63, no. 6, pp. 1664–1675, 2018.

[22] D. Pickem, P. Glotfelter, L. Wang, M. Mote, A. Ames, E. Feron, and M. Egerstedt, "The Robotarium: A remotely accessible swarm robotics research testbed," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 1699–1706.

[23] S. Cho, V. Mishra, Q. Tao, P. Vamell, M. King-Smith, A. Muni, W. Smallwood, and F. Zhang, "Autopilot design for a class of miniature autonomous blimps," in *2017 IEEE Conference on Control Technology and Applications (CCTA)*, 2017, pp. 841–846.

[24] W. L. Scott and N. E. Leonard, "Optimal evasive strategies for multiple interacting agents with motion constraints," *Automatica*, vol. 94, pp. 26 –34, 2018.

[25] M. R. Brust, G. Danoy, P. Bouvry, D. Gashi, H. Pathak, and M. P. Gonalves, "Defending against intrusion of malicious uavs with networked uav defense swarms," in *Proceedings of IEEE Conference on Local Computer Networks (IEEE LCN)*, 2017, pp. 103–111.

[26] P. Ogren, E. Fiorelli, and N. E. Leonard, "Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment," *IEEE Transactions on Automatic Control*, vol. 49, no. 8, pp. 1292–1302, 2004.

[27] L. Arranz, L. Schenato, and A. Seuret, "Distributed source seeking via a circular formation of agents under communication constraints," *IEEE Transactions on Control of Network Systems*, vol. 3, no. 2, pp. 104 –115, 2016.

[28] R. Fabbiano, F. Garin, and C. Canudas-de-Wit, "Distributed source seeking without global position information," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 1, pp. 228–238, 2018.

[29] K. Sakurama and S. Nishida, "Source seeking by distributed swarm robots with sample variance control," in *Proceedings of of American Control Conference*, 2016, pp. 2484 –2487.

[30] W. Wu and F. Zhang, "A speeding-up and slowing-down strategy for distributed source seeking with robustness analysis," *IEEE Transactions on Control of Network Systems*, vol. 3, no. 3, pp. 231 –240, 2016.

[31] W. Wu, D. Chang, and F. Zhang, "A bio-inspired robust 3d plume tracking strategy using mobile sensor networks," in *Proceedings of 52nd IEEE Conference on Decision and Control*, 2013, pp. 4571–4578.

[32] K. B. Ariyur and M. Krstic, *Real-Time Optimization by Extremum-Seeking Control*. Wiley Interscience, 2003.

[33] J. Cochran and M. Krstic, "Nonholonomic source seeking with tuning of angular velocity," *IEEE Transactions on Automatic Control*, vol. 54, no. 4, pp. 717–731, 2009.

[34] J. Cochran, A. Siranosian, N. Ghods, and M. Krstic, "3D source seeking for underactuated vehicles without position measurement," *IEEE Transactions on Robotics*, vol. 25, no. 1, pp. 117–129, 2009.

[35] S. Z. Khong, Y. Tan, C. Manzie, and D. Nei, "Multi-agent source seeking via discrete-time extremum seeking control," *Automatica*, vol. 50, no. 9, pp. 2312 –2320, 2014.

[36] K. Kvaternik and L. Pavel, "An analytic framework for decentralized extremum seeking control," in *2012 American Control Conference (ACC)*, 2012, pp. 3371–3376.

[37] Y. Elor and A. M. Bruckstein, "robot cloudi gradient climbing with point measurements," *Theoretical Computer Science*, vol. 547, pp. 90 –103, 2014.

[38] E. Luo, J. Liu, A. Bacula, Y. Ng, T. Basar, and G. X. Gao, "Shinerbots: Navigating large-scale robot swarms inspired by golden shiner fish," *Swarm Intelligence (Submitted)*,

[39] A. Turgeman and H. Werner, "Mission control - combined solutions for source seeking and level curve tracking in a time-varying field," in *2017 American Control Conference (ACC)*, 2017, pp. 4268–4273.

[40] S. A. Barogh and H. Werner, "Cooperative source seeking with distance-based formation control and non-holonomic agents," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 7917 –7922, 2017, 20th IFAC World Congress.

[41] S. Chatterjee and W. Wu, "Cooperative curve tracking in two dimensions without explicit estimation of the field gradient," in *2017 4th International Conference on Control, Decision and Information Technologies (CoDIT)*, 2017, pp. 0167–0172.

[42] K. Ovchinnikov, A. Semakova, and A. Matveev, "Decentralized multi-agent tracking of unknown environmental level sets by a team of nonholonomic robots," in *2014 6th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, 2014, pp. 352–359.

[43] A. S. Matveev, M. C. Hoy, K. Ovchinnikov, A. Anisimov, and A. V. Savkin, "Robot navigation for monitoring unsteady environmental boundaries without field gradient estimation," *Automatica*, vol. 62, pp. 227 –235, 2015.

[44] D. T. Swain, I. D. Couzin, and N. E. Leonard, "Coordinated speed oscillations in schooling killifish enrich social communication," *Journal of Nonlinear Science*, vol. 25, no. 5, pp. 1077–1109, 2015.

[45] G. Wagner and H. Choset, "Gaussian reconstruction of swarm behavior from partial data," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 5864–5870.

[46] M. Haque, E. Baker, C. Ren, D. Kirkpatrick, and J. Adams, "Analysis of biologically inspired swarm communication models," vol. 85, pp. 18–38,

[47] D. Mateo, Y. K. Kuan, and R. Bouffanais, "Effect of correlations in swarms on collective response," *Nature*, vol. 7, no. 10388, 2017.

[48] M. Mischiati and P. S. Krishnaprasad, "Geometric decompositions of collective motion," *Proceedings of the Royal Society A*, vol. 473, 2017.

[49] A. Jadbabaie, N. Motee, and M. Barahona, "On the stability of the kuramoto model of coupled nonlinear oscillators," in *Proceedings of the 2004 American Control Conference*, vol. 5, 2004, 4296–4301 vol.5.

[50] N. Moshtagh and A. Jadbabaie, "Distributed geodesic control laws for flocking of nonholonomic agents," *IEEE Transactions on Automatic Control*, vol. 52, no. 4, pp. 681 –686, 2007.

[51] C. Lageman and Z. Sun, "Consensus on spheres: Convergence analysis and perturbation theory," in *Proceedings of 55th IEEE Conference on Decision and Control*, 2016, pp. 19 –24.

[52] E. D. Sontag, "On the input-to-state stability property," *European Journal of Control*, vol. 1, no. 1, pp. 24 –36, 1995.

[53] V. K. H. Tanner and G. Pappas, "Stability properties of interconnected vehicles," in *Proceedings of 15th International Symposium of Mathematical Theory of Networks and Systems*, 2002.

[54] I. T. Jollifi, *Principal Component Analysis*. Springer, 2002.

[55] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 4, pp. 433–459,

[56] W. Yong, U. Helmke, and J. B. Moore, "Global analysis of Oja's flow for neural networks," *IEEE Transactions on Neural Networks*, vol. 5, no. 5, pp. 674–683, 1994.

[57] L. Xu, "Least mean square error reconstruction principle for self-organizing neural-nets," *Neural Networks*, vol. 6, no. 5, pp. 627 –648, 1993.

[58] J. H. Manton, U. Helmke, and I. M. Y. Mareels, "Dynamical systems for principal and minor component analysis," in *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*, vol. 2, 2003, pp. 1863–1868.

[59] M. A. Hasan, "Stability analysis of dynamical systems for minor and principal component analysis," in *2006 American Control Conference*, 2006, 6 pp.–.

[60] I. D. Couzin, J. Krause, N. R. Franks, and S. A. Levin, "Effective leadership and decision-daking in animal groups on the move," *Nature*, vol. 433, pp. 513–516, 2005.

[61] M. M. Webster, "Experience and motivation shape leaderfollower interactions in fish shoals," *Behavioral Ecology*, vol. 28, no. 1, pp. 77–84, 2017.

[62] E. W.J. O. R. Zou V. Kalivarapu and S. Bhattacharya, "Particle swarm optimization-based source seeking," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 3, pp. 865 –875, 2015.

[63] M. Fahad, Y. Guo, B. Bingham, K. Krasnosky, L. Fitzpatrick, and F. A. Sanabria, "Evaluation of ocean plume characteristics using unmanned surface vessels," in *OCEANS 2017 - Anchorage*, 2017, pp. 1–7.

[64] A. Flack, M. Nagy, W. Fiedler, I. D. Couzin, and M. Wikelski, "From local collective behavior to global migratory patterns in white storks," *Science*, vol. 360, no. 6391, pp. 911–914, 2018.

[65] T. Ibuki, T. Hatanaka, M. Fujita, and M. W. Spong, "Visual feedback pose synchronization with a generalized camera model," in *2011 50th IEEE Conference on Decision and Control and European Control Conference*, 2011, pp. 4999–5004.

[66] E. W. Justh and P. S. Krishnaprasad, "Natural frames and interacting particles in three dimensions," in *Procedings of 44th IEEE Conference on Decision and Control and the European Control Conference*, IEEE, 2005, pp. 2841–2846.

[67] W. Wu and F. Zhang, "Cooperative exploration of level surfaces of three dimentional scalar fields," *Automatica*, vol. 47(9), pp. 2044–2051, 2011.

[68] H. K. Khalil, *Nonlinear Systems, Third Edition*. Prentice Hall, 2002.

[69] D. C. K. B. C. Daniels C. J. Ellison and J. C. Flack, "Quantifying collectivity," *Elsevier*, vol. 37, pp. 106–113, 2016.

[70] P. Glotfelter, I. Buckley, and M. Egerstedt, "Hybrid nonsmooth barrier functions with applications to provably safe and composable collision avoidance for robotic systems," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1303–1310, 2019.

[71] F. Zhang and N. E. Leonard, "Cooperative filters and control for cooperative exploration," *IEEE Transactions on Automatic Control*, vol. 55, no. 3, pp. 650–663, 2010.

[72] R. K. Williams and G. S. Sukhatme, "Probabilistic spatial mapping and curve tracking in distributed multi-agent systems," in *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 1125–1130.

[73] A. Strandburg-Peshkin, C. R. Twomey, N. W. F. Bode, A. B. Kao, Y. Katz, C. C. Ioannou, S. B. Rosenthal, C. J. Torney, H. S. Wu, S. A. Levin, and I. D. Couzin, "Visual sensory networks and effective information transfer in animal groups," *Current Biology*, vol. 23, no. 17, R709–R711, 2013.

[74] W. L. Romey, "Individual differences make a difference in the trajectories of simulated schools of fish," *Ecological Modelling*, vol. 92, no. 1, pp. 65 –77, 1996.

[75] J. W. Jolles, N. J. Boogert, V. H. Sridhar, I. D. Couzin, and A. Manica, "Consistent individual differences drive collective behavior and group functioning of schooling fish," *Current Biology*, vol. 27, no. 18, 2862 –2868.e7, 2017.

[76] I. P. Sitj, S. C. Nicolis, G. Sempo, and J.-L. Deneubourg, "The interplay between personalities and social interactions affects the cohesion of the group and the speed of aggregation," *PLOS ONE*, vol. 13, no. 8, Aug. 2018.

[77] J. W. Jolles, K. L. Laskowski, N. J. Boogert, and A. Manica, "Repeatable group differences in the collective behaviour of stickleback shoals across ecological contexts," *Proceedings of the Royal Society of London B: Biological Sciences*, vol. 285, no. 1872, 2018.

[78] D. Bierbach, T. Landgraf, P. Romanczuk, J. Lukas, H. Nguyen, M. Wolf, and J. Krause, "Using a robotic fish to investigate individual differences in social responsiveness in the guppy," *bioRxiv*, 2018.

[79] T. Schaerf, J. Herbert-Read, M. R. Myerscough, D. Sumpter, and A. Ward, "Identifying differences in the rules of interaction between individuals in moving animal groups," Jan. 2016.

VITA

Said Al-Abri is currently a Ph.D. candidate at the School of Electrical and Computer Engineering at the Georgia Institute of Technology, expecting to graduate in August 2019. He earned two M.S. degrees in Electrical and Computer Engineering from the Georgia Institute of Technology in 2017 and from the University of Central Florida in 2013. He earned a B.S. degree in Electrical and Computer Engineering from the Sultan Qaboos University, Oman, in 2009. His research focus is on bio-inspired learning and control for multi-agent control systems. He is currently a member of the Georgia Tech Systems Research Laboratory where he is working on designing and implementing bio-inspired algorithms that enable efficient information propagation across robotic swarms. He has teaching experience at Georgia Tech and Sultan Qaboos University.